# Monte Carlo studies on some thermal properties of one-dimensional system

# 一維系統熱力學性質之蒙地卡羅研究

by

Chan Kwok Ming

（陳國明）

A Thesis Submitted In Partial Fulfillment
of the Requirements for the Degree of
Master of Philosophy in Physics

The Chinese University of Hong Kong

June 1987

CONTENTS

Acknowledgement

I would like to express my gratitude towards my supervisors, Dr. K.L. Liu and Prof. K. Young, for their guidances and encouragements during my work and the preparation of this thesis.

I would also like to thank Mr. K.Y. Chan for his kind approval of using some of his results and Mr. Y.K. Ng for his help in typing this thesis.

## Abstract

The Monte Carlo method is used to study some thermal properties of the 1-dimensional fermion system. Some modifications of the Monte Carlo method proposed by Hirsch and Sugra are also suggested. The modified method may be useful not only in solving the problems of the fermion systems but also in solving other classical and quantum problems using the Monte Carlo method. Small beta expansion is also discussed. The modified Monte Carlo simulation in conjunction with the Padé approximant canprovide some useful estimates at low temperatures.

## List of Figures

# Chapter 1

# Introduction to the Monte Carlo method

## 1.1 General introduction

In statistical mechanics one wants to calculate thermodynamic properties of many-body systems. Most of these systems cannot be solved exactly, so many approximation techniques have been developed. For example, with the systematic language of Feynman rules and diagrams, physical quantities can in principle be evaluated in perturbation theory to any finite order. Another successful method is the low- and high-temperature expansions, applied for example to the 3-D Ising model. One systematic non-perturbative method is the Monte Carlo Method. In classical statistical mechanics the application of the Monte Carlo method is straightforward (Binder, 1979). Configurations in configuration space are sampled with an importance-sampling technique (often called the Metropolis method) using the classical Boltzmann probability measure and the "time" averages of the thermodynamic quantities over the length of the "experiment" then provide estimates of the corresponding ensemble averages.

The success of the Monte Carlo method for classical systems raises the question of its applicability to quantum statistical mechanics.

## 1.2 Classical Monte Carlo Method

Consider a classical system with say three discrete states. If the energies of the states are $E_1$, $E_2$ and $E_3$ respectively, and the probabilities (at a certain temperature) of finding the particle in each state are $P_1$, $P_2$, and $P_3$ respectively, then the average energy of the system is given by ensemble average

$$E = \frac{P_1*E_1+P_2*E_2+P_3*E_3}{P_1+P_2+P_3}$$

We can solve the problem from another point of view. We can imagine that the particle is hopping between the states. The number of times that the particle visits a state $S_i$ is proportion to the probability $P_i$. In equilibrium, the rate of hopping out of state $S_i$ must equal the rate of hopping into $S_i$, i.e.

$$P_i \sum_j R_{ij} = \sum_j P_j*R_{ji}$$

where the summation is over all states, $R_{ij}$ is the probability of a particle hopping from state i to state j.

The above equation is the "global balance" condition. With a stronger constraint, we assume that the detailed-balance condition holds; i.e. the number of the particle hopping from state $S_i$ to $S_j$ is equal to the number of hopping from state $S_j$ to $S_i$. The number of hopping from state $S_i$ to $S_j$ depends on two factors. One is the probability of finding the particle in state $S_i$ ($P_i$),

another is the probability of hopping to $S_j$ if the particle is in state $S_i$ ($R_{ij}$). The number of hopping from $S_i$ to $S_j$ is equal to the product of the two factors, $P_i * R_{ij}$. When the detailed-balance holds,

$$P_i * R_{ij} = P_j * R_{ji}$$

From the above equation, $R_{ij}$ should be expressible in terms of the equilibrium values of $P_i$ and $P_j$, but there is a lot of possible forms for the dependence. We may choose the following simple form for our simulation,

$$R_{ij} = \frac{P_j}{P_i + P_j} \qquad (1.1)$$

This choice will satisfy the detailed-balance condition.

The energy of the system is the ensemble average of the energy of the states and is also equal to the time average of the energy of the particle. At time zero, the particle may be in state $S_1$, then goes to state 2, and moves around the states. The different probabilities of finding the particle in different states are simulated by the following scheme:

(1) Choose a state as the current state, $S_i$

(2) Choose a state as the next state, $S_j$

(3) Define hop rate $R = P_j/(P_i+P_j)$

(4) Generate a random number r, $0<r<=1$

(5) If $r<R$, accept $S_j$ as the current state; otherwise keep the $S_i$ as current state.

(6) Repeat steps (2)-(5)

With the above scheme, the detailed-balance condition is guaranteed to hold. The particle will visit the states according to the state-probabilities, and the time average of the energy of the particle is equal to the energy of the system if the number of hopping steps is large enough.

There is still the problem of choosing the next state at each step. Each state should have the same probability of being chosen as the candidate for the next state. In many procedures of choosing the next state, this requirement may not be easy to fulfil. This difficulty will be treated in Chapter 2.

One may prefer taking the ensemble-average energy directly to using the above method. This is practicable only when the number of states is not too large.

In a large number of many-body problems, a macroscopic system may be reduced to a rather small system containing only a few particles, for example less than 10. This is due to the fact that the correlation length is usually much smaller than the actual size of the real system. For example, a dilute system may contain $10^{20}$ atoms, but typically, only 10 atoms may have interaction at a time, i.e. the correlation length is about the length L such that in $L^3$ there are 10 atoms. In this case, we may use a 10-particle system to simulate the real system. This will reduce the system size to a much smaller value. However, the

number of the eigenstates is still too large to cope with, because it may be proportional to the exponential of the number of particle.

For example, if ten particles are put in a 2-D 10*10 lattice, one to a site, then the number of configurations (or states) is 100!/10! $\approx 10^{100}$. This is a system too large for direct averaging, then the importance sampling scheme may be useful. From experience, usually the number of simulation steps needed is much smaller than the number of total states. This will be shown in the following Chapters.

The Metropolis Monte Carlo method (Metropolis et al., 1953) is a particularly simple but very powerful algorithm of importance sampling. One can prove that the sequence of states generated by this algorithm is a Markov chain of which the limiting distribution is according to the state probabilities.

In practical applications, the most important condition that should be fulfilled is that of ergodicity. This means that in selecting a trial state S one has to be sure that every allowed state of the system may be reached.

## 1.3 Quantum Monte Carlo method

In quantum mechanical problems, one would need to calculate the average of operators

$$\langle A \rangle = \frac{tr\ (Ae^{-\beta H})}{tr\ (e^{-\beta H})}$$

where H is the Hamiltonian.

Usually it is impossible to solve the eigenvalues of the Hamiltonian of the interesting many-body system exactly, because the dimension of the Hilbert space, i.e. the number of eigenstates, is too large for even the largest computer used nowadays. Consider a 20-site-10-fermion system as an example. If only one fermion is allowed at a site, then the number of eigenstates (or the dimension of the Hilbert space) is equal to the number of possible distributions of the 10 fermions, that is $20!/10!*10! \approx 2*10^5$. In order to calculate the determinant using the Gauss elimination method, there are $(2*10^5)^3/3 \approx 10^{15}$ divisions and multiplications, which are usually performed in real (floating point) arithmetic. It is easily seen that this is not calculable even for a computer which can calculate 1 billion multiplications per second. (Arfken, G.)

In nearly all solvable cases that we know of, the Hamiltonian of the interacting many-body system is a sum of several operators that can be diagonalized separately. This suggests that one should try to construct systematic approximations to $e^{-\beta H}$ by using the knowledge of the spectrum of the operators that contribute to H. The Trotter formula (Trotter, 1959) and its generalizations (Suzuki,

1976) provide the necessary mathematical justification for the construction of the approximations.

# Chapter 2

## One-dimensional fermion system

### 2.1 Introduction

Studies of fermion systems in one dimension are of great interest from the theoretical point of view. A lot of many-body problems associated with the electron systems and spin systems can be related to the fermion systems. Problems are frequently considered in one dimension because of its relative simplicity. Moreover, the existence of quasi-one-dimensional conductors and quasi-one-dimensional magnets gives further physical significance to the studies of one-dimensional models.

For electronic systems, the number of fermions is conserved and there can be only zero or one fermion at a site if the spin degree of freedom is neglected. Spin one-half systems on a one-dimensional lattice can always be transformed to fermion systems through Jordan-Wigner's transformation (P. Jordan and E. Wigner). For a magnetic system, the magnetization is not conserved. This makes the dimension of the Hilbert space of a magnetic system much larger than that of an electronic system. Application of the Monte Carlo method to one-dimensional fermion systems may be found in Hirsch (1982).

A one-dimensional fermion system with nearest-neighbor interactions can be described by the Hamiltonian

$$H = \sum H_{i,i+1} \qquad \text{for } i=1\ldots N$$

$$H_{i,i+1} = -t\,(C_i^{+}C_{i+1} + C_{i+1}^{+}C_i) + V\,n_i\,n_{i+1}$$

where $C_i^{+}$ and $C_i$ are the creation and annihilation operators for a fermion at the ith spatial lattice site, satisfying $\{C_i^{+}, C_j\}=\delta_{ij}$, and $n_i=C_i^{+}C_i$ is the number operator. Summation is from 1 to number of sites, N; and periodic boundary condition is usually applied, i.e. site N+1 is identified with site 1.

The first term of the Hamiltonian is the hopping energy of the fermions, and t is the hopping constant. The second term is the nearest neighbor interaction. A positive interaction constant V means the fermions repel. A constant N*V/4 will be added to the second term of the Hamiltonian in order to simplify the calculation. The modified Hamiltonian is

$$H = \sum H_{i,i+1} \qquad \text{for } i=1\ldots N$$

$$H_{i,i+1} = -t\,(C_i^{+}C_{i+1} + C_{i+1}^{+}C_i) + V\,(n_i-1/2)(n_{i+1}-1/2) \qquad (2.1)$$

Note that this is only true for half-filled system. We will use this Hamiltonian in the following chapters.

## 2.2 Trotter-formula approach

The system we are considering is a N-site system. As usual, we want to relate this system to a combination of

two-site systems since the latter is much easier to cope with. The first difficulty that we meet is the non-commutability of the terms in the Hamiltonian, so that we cannot break the system into two-site systems directly. We cannot say, for example, write

$$\exp\Big(\sum_i H_{i,i+1}\Big) = \prod_i \exp(H_{i,i+1})$$

because $H_{i,i+1}$ does not commute with $H_{i+1,i+2}$.

In order to apply the Monte Carlo Method, let us start by rewriting the Hamiltonian in the form

$$H = H_a + H_b,$$

where $H_a = \Sigma H_{i,i+1}$, for $i = 1,3,\ldots N-1$

$H_b = \Sigma H_{i,i+1}$, for $i = 2,4\ldots N$

$$H_{i,i+1} = -t\,(C_i^{+}C_{i+1} + C_{i+1}^{+}C_i) + V\,(n_i - 1/2)(n_{i+1} - 1/2) \qquad (2.2)$$

N, taken to be even, is the size of the system.

Notice that $H_a$ and $H_b$ are each composed of a sum of N/2 mutually commuting terms.

So, for example, we may write

$$\exp(-\beta H_a) = \prod \exp(-\beta H_{i,i+1}) \qquad i = 1,3,\ldots N-1.$$

It is useful to think of $\beta$ as an imaginary time $\beta = it$, so that the Boltzmann factor $\exp(-\beta H)$ can be thought of as the time-evolution operator $\exp(-iHt)$, for the evaluation of which we have the well known path-integral technique (Feynman).

As in the standard derivation of path integrals, we begin by dividing the imaginary-time interval $0<\tau<\beta$ into m subintervals of width $\Delta\tau=\beta/m$. At each time slice we insert a complete set of states so that the partition function, for example, is given by

$$Z=tr(e^{-\beta H}) = \sum \langle S_1|e^{-\beta H/m}|S_2\rangle\langle S_2|e^{-\beta H/m}|S_3\rangle\ldots\langle S_{m-1}|e^{-\beta H/m}|S_m\rangle \quad (2.3)$$

where the summation is over all states and $|S_m\rangle=|S_1\rangle$.

In our problem, the Hamiltonian can be rewritten as follows:

$$H = H_a + H_b$$

with $H_a$ and $H_b$ each being easily diagonalized. Then with the use of the fact that for small $\Delta\tau$,

$$\exp(-\Delta\tau H) = \exp(-\Delta\tau H_a)*\exp(-\Delta\tau H_b)*[1 + O(\Delta\tau)^2]$$

and inserting additional intermediate states we have approximately

$$Z = \sum \langle S_1|U_a|S_2\rangle\langle S_2|U_b|S_3\rangle\ldots\langle S_{2m-1}|U_b|S_{2m}\rangle \quad (2.4)$$

where $U_a=\exp(-\beta H_a/m)$, and $U_b=\exp(-\beta H_b/m)$.

By a judicious choice of the complete sets of intermediate states, all the matrix elements in (2.4) can be directly evaluated if the number of sites and m are small.

However, when the number of sites and m are not small, it will take a lot of time to go through all the possible

states. To overcome this difficulty, the importance sampling of the Monte Carlo approach can be used.

## 2.3 Checkerboard representation

We note that within each time interval $\Delta\tau$ there is one application of the operator $U_a$ and one of the operator $U_b$. Since $H_a$ and $H_b$ only contain mutually commuting terms, this leads to the graphical representation shown in Fig. 1.

$$\begin{aligned}
&\langle S|\exp(-\beta H_a/m)|S'\rangle \\
&=\langle S|\exp(-\beta H_{1,2}/m)\ \exp(-\beta H_{3,4}/m)..\ \exp(-\beta H_{N-1,N}/m)|S'\rangle \\
&=\langle S_{1,2}|\exp(-\beta H_{1,2}/m)|S'_{1,2}\rangle\ \langle S_{3,4}|\exp(-\beta H_{3,4}/m)|S'_{3.4}\rangle \\
&.....\langle S_{N-1,N}|\exp(-\beta H_{N-1,N}/m)|S'_{N-1,N}\rangle \qquad (2.5)
\end{aligned}$$

where $|S_{i,i+1}\rangle$ is a state of two-site problem. There are four possible states of $|S_{i,i+1}\rangle$: they are, in terms of occupation numbers on the two sites, $|0,0\rangle$, $|0,1\rangle$, $|1,0\rangle$, and $|1,1\rangle$.

Here the periodic spatial lattice of sites is labeled by n and the imaginary-time axis $\tau$ has been sliced into $2\beta/\Delta\tau$ segments. The occupation on each slice corresponds to one of the states $|S_k\rangle$ in the sum for the partition function Z. The un-shaded boxes correspond to the areas of space and imaginary time in which fermions can hop and interact. The sum over intermediate states corresponds to the sum over all possible ways of distributing the fermions on the spatial lattice at each time slice. So any summation over all

intermediate states is equivalent to summation over all allowed checker-board configurations.

If in the checker-board representation, $\langle S|e^{-\beta H}|S\rangle$ is denoted by P(S), then the partition function is given by

$$Z = \sum_S \langle S|e^{-\beta H}|S\rangle = \sum_S P(S)$$

where summation is over all states.

In performing the summation over fermion configurations by importance sampling, we generate new configurations and accept or reject them according to the detailed-balance condition mentioned in chapter 1.2, which ensures that the probability of a configuration being sampled is proportional to the weighting factor in the summation. In generating new configurations, the first constraint we must take into account is the conservation laws associated with the Hamiltonian, i.e., the total number of fermions must be N/2 in each state; otherwise a large amount of computer time will be wasted in generating configurations that have zero probability of being accepted. In other words, we will generate an allowed state directly instead of generating a series of states and then determine whether they are allowed configurations.

Fermion number is always conserved. With our breakup procedure it is conserved by each $H_{i,i+1}$, that is within each un-shaded box in Fig. 1. The occupied sites at each $\tau$ slice have been connected by lines which we will call the world lines of the fermions. The summation over intermediate

states that satisfy fermion-number conservation is equivalent to the summation over all allowed configurations of the world lines. Notice that world lines can be drawn along the vertical edge of a un-shaded box (s type) or diagonally across a un-shaded box (d type), but they cannot be drawn diagonally across a shaded box. Four corners that are fully occupied (f type) or empty (n type) are also allowed configurations. These four types of basic blocks are shown in Fig. 2. For those disallowed types of boxes, the probabilities are zero. The probabilities of the allowed types of unshaded boxes are as follows:

$$
\begin{aligned}
Ps &= \langle 1,0|\exp(-\beta H_{1,2}/m)|1.0\rangle \\
&\quad (\text{or } = \langle 0,1|\exp(-\beta H_{1,2}/m)|0,1\rangle) \\
&= \exp(\beta V/2m)\ \mathrm{Cosh}(\beta t/m) \\
Pd &= \langle 1,0|\exp(-\beta H_{1,2}/m)|0,1\rangle \\
&\quad (\text{or } = \langle 0,1|\exp(-\beta H_{1,2}/m)|1,0\rangle) \\
&= \exp(\beta V/2m)\ \mathrm{Sinh}(\beta t/m) \\
Pf &= \langle 1,1|\exp(-\beta H_{1,2}/m)|1.1\rangle \\
&= \exp(-\beta V/2m) \\
Pn &= \langle 0,0|\exp(-\beta H_{1,2}/m)|0.0\rangle \\
&= \exp(-\beta V/2m)
\end{aligned}
\qquad (2.6)
$$

The probability of a checker-board state is given by the the product of all the probabilities of the un-shaded boxes. $P(S) = \pi\ P(S_i)$ where $P(S_i)$ is the probability of the ith un-shaded box. If the number of s, d, f and n-type un-shaded boxes are Ns, Nd, Nf and Nn respectively, then the probability of that configuration is

$$P = Ps^{Ns} * Pd^{Nd} * Pf^{Nf} * Pn^{Nn}$$

All the thermal properties can be calculated starting from the probability of the configuration. For example, the internal energy of the state is related to the first derivative of the probability with respect to beta.

We may start from the following equation:

$$E = \frac{-1}{Z} \frac{\partial Z}{\partial \beta}$$

where partition function $Z = \sum P(S)$

and $P(S) = \pi_i P(S_i)$

so,

$$E = \frac{-1}{\sum P(S)} \sum \frac{\partial}{\partial \beta} \pi_i P(S_i)$$

$$E = \frac{-1}{\sum P(S)} \sum P(S) \sum_i \frac{1}{P(S_i)} \frac{\partial}{\partial \beta} P(S_i)$$

define $E(S_i)$ as follows:

$$E(S_i) = \frac{-1}{P(S_i)} \frac{\partial}{\partial \beta} P(S_i)$$

and

$$E(S) = \sum_i E(S_i)$$

then we have

$$Es = -\frac{V}{2m} - \frac{t}{m}\,\mathrm{Tanh}\left(\frac{\beta t}{m}\right)$$

$$Ed = -\frac{V}{2m} - \frac{t}{m}\,\mathrm{Coth}\left(\frac{\beta t}{m}\right)$$

$$Ef = \frac{V}{2m}$$

$$En = \frac{V}{2m}$$

and

$$E = \frac{\sum P(S)E(S)}{\sum P(S)}$$

For heat capacity C with $k_B$=1, we have

$$C = -\beta^2 \frac{\partial E}{\partial \beta}$$

$$C = -\beta^2 \frac{\partial}{\partial \beta} \frac{\sum P(S)E(S)}{\sum P(S)}$$

$$C = -\beta^2 \frac{1}{[\sum P(S)]^2} \left[ \sum P(S)E(S) \sum P(S)E(S) + \sum P(S) \sum P(S)C(S) - \sum P(S) \sum P(S)E(S)^2 \right]$$

where $C(S_i)$ is defined as follows:

$$C(S_i) = \frac{\partial}{\partial \beta} E(S_i)$$

and also

$$C(S) = \sum C(S_i)$$

and

$$Cs = -\left(\frac{t}{m}\right)^2 \mathrm{Sech}\left(\frac{\beta t}{m}\right)^2$$

$$Cd = \left(\frac{t}{m}\right)^2 \mathrm{Csch}\left(\frac{\beta t}{m}\right)^2$$

$$Cn = 0$$

$$Cf = 0$$

We may use the following notations in short:

$$\langle E(S)\rangle_{mc} = \frac{\sum P(S)E(S)}{\sum P(S)}$$

$$\langle C(S)\rangle_{mc} = -\beta^2 \frac{\partial}{\partial\beta} \langle E(S)\rangle_{mc}$$

then

$$E = \langle E(S)\rangle_{mc}$$

$$C = -\beta^2\langle C(S)\rangle_{mc} - \beta^2[\langle E(S)\rangle_{mc}^2 - \langle E(S)^2\rangle_{mc}]$$

Note that all the thermal quantities depend on the number of the different types of the un-shaded boxes, but do not depend on the spatial distribution of the boxes.

Some people may note that the $P(S_i)$'s have been thought of as appearing probabilities of the configurations. But, in fact, depending on the sign of t, Pd can have negative values and thus it should not be considered as a probability. In all of our discussions, only P(S) should have the meaning of probabilities, but not $P(S_i)$. Note that in any configurations, the number of type d un-shaded box is

even, i.e. P(S) must be positive; so P(S) can be thought of the appearing probability of that configuration.

## 2.4 Algorithm for generating states

We wish to develop an algorithm which should allow us to go through the whole allowed configuration space, or at least go through the main part of the space.

Although we can generate configurations by putting the allowed un-shaded boxes together one by one, and then checking the checker-board after each step, the whole process is too time-consuming because only a small portion of the configurations so generated are allowed states. There is neither a simple algorithm to build an allowed configuration directly nor a simple way to determine whether a checker-board configuration is an allowed state or not. The only method is to check the whole checker-board step by step. It is also time-consuming to calculate the probability of a configuration even if the configuration is an allowed state, since we need to count the numbers of different types of un-shaded boxes in the whole checker-board. (Note that an N*N checker-board has $N^2/2$ un-shaded boxes.) What we want is an algorithm that can build a new configuration by only changing a small part of the old configuration. Since the old configuration is an allowed state, the unchanged parts should give no problem. We only need to take care of the changed parts and their connections to the unchanged parts. Since the probability of a checker-board configuration

depends only on the number of different types of unshaded boxes, the probability change also depends only on the four un-shaded boxes connected to the chosen shaded box. The characteristic of the algorithm mentioned above is locality. Not only are the changes of the checker-board local, but also the changes of the probabilities. This character of the algorithm makes the configuration generation simple and the calculation of the probability easy. How much time will be saved depends on the size of the checker-board; for a large checker-board, much time will be saved.

In order to determine the hop rate, we are only interested in the ratio of probabilities before and after hopping. The hop rate is given by

$$R = \frac{P(old)}{P(old)+P(new)} = \frac{1}{1 + P(new)/P(old)}$$

If the increase of Ns, Nd, Nf and Nn are $\Delta$Ns, $\Delta$Nd, $\Delta$nf and $\Delta$Nn respectively, then we have

$$\frac{P(new)}{P(old)} = Ps^{\Delta Ns} * Pd^{\Delta Nd} * Pf^{\Delta Nf} * Pn^{\Delta Nn}$$

2.4.1 Two-particle jump procedure

We wish to develop an algorithm for generating all allowed world-line configurations by only changing a part of the old configuration. The smallest change one may think of

is moving a fermion to a site next to its current site. But this will always lead to fermion non-conservation in at least one unshaded box, or in other words, break the fermion world-line. Instead of a single site, we may choose a shaded box as the change-part. Not all the shaded boxes can be chosen for this change; only the shaded boxes with two fermions on one vertical edge may be chosen for the change. For example, if the shaded box chosen has only a single fermion in the upper left corner, and if we change the position of the fermion, this will break the fermion world-line. Although we may compensate by changing the shaded boxes next to it, this will make the procedure much more complex. (In fact, the compensation may involve many changes in boxes indirectly connected to the chosen shaded box, this will give us a lot of trouble.)

The minimum change we can make is to move two fermions from one vertical edge of a shaded box to another vertical edge as illustrated in Fig. 3. It can be easily shown that the configurations of the four unshaded boxes connected to the chosen shaded box are still allowed types no matter what their current types are. In order to calculate the probability of the new configuration, we only need to handle the probabilities of the four unshaded boxes since these are the only changes in the whole configuration. The procedure may be summarized as follows:

(1) Choose a shaded box randomly.

(2) If two fermions are on one side of the box, move the two fermions to the other side; otherwise repeat step (1).

(3) Update the four unshaded boxes connected to the shaded box. The probability of the new configuration depends on the changes of the four un-shaded boxes and the probability of the old configuration.

We will carry out some steps as demonstration. Suppose at time zero, the checker board is as shown in Fig. 3.

(1) Choose shaded box number 2 (this choice is random).

(2) Since there is only one fermion at the corner of the box, we reject this choice.

(3) Choose shaded box number 3 (this choice is also random).

(4) Since there are two fermions on one side of the box, we accept this choice.

(5) Generate a new configuration by moving the fermions to the other side.

(6) Check the changes of the four un-shaded boxes. In this example, $\Delta N_s=2, \Delta N_d=0, \Delta N_f=\Delta N_n=-1$.

(7) The hop rate is

$$R = \frac{1}{1 + P_s^{2} * P_f^{-1} * P_n^{-1}}$$

(8) Generate a random number r, $0<r<=1$; if $r<R$, accept the new configuration as the next state; otherwise keep the old configuration.

(9) Choose another shaded box and repeat the above steps.

Following the above procedure, we can generate a series of allowed checker-board configurations.

2.4.2 Winding number

In keeping track of the checker-board configuration, it is useful to divide the configurations into different classes. The winding number provides one important scheme of classification.

We define the winding number of a configuration in the following manner. Because we are evaluating a trace, the boundary condition in the "vertical" (i.e. time) direction must be periodic. Moreover, we use periodic boundary conditions in the "horizontal" (i.e. spatial) direction, so a checker-board has the topology of a torus. (Another choice of the boundary condition of the spatial (horizontal) direction is fixed-boundary.) Suppose we start at any occupied site at time $\tau=0$ and follow a world line continuously through one complete evolution. Note that the top and the bottom of a checker-board have the same configuration, and a fermion at the top connects to a fermion at the bottom of the checker-board by what we call the fermion world line. If the ith fermion at the top connects to the ith fermion at the bottom by its world line, then the winding number of that checker-board configuration is called zero. Note that if one of the fermion at the top connects to itself in the manner describe above, then all the fermions at the top will do so. If the ith fermion at the top connects to the (i+n)th fermion at the bottom, then the winding number of that configuration is n. (Note that for a N-site system, if the winding number is 1, then the

Nth fermion at the top connects to the first fermion at the bottom since the checker-board has the topology of a torus.) Examples of configurations with winding number zero and one are shown in Fig. 4.

In most applications, for all but the smallest spatial lattices (i.e., so long as the length of the fermion system N>6), it is believed that only configurations with winding number 0 are important (Hirsch et al 1982). Configurations with non-zero winding number only arise when periodic boundary conditions are imposed in the spatial direction and do not occur when fixed boundary conditions are imposed.

Since the two-particle jump procedure preserves winding number it alone is inadequate for generating all possible configurations. In order to generate new configuration with a different winding number, two methods can be used. One basic method is to randomly generate one new allowed configuration with different winding number. The disadvantage is that it is very time consuming to generate the new configuration and calculate its probability. This makes the method not applicable in practice. Another method is the N-particle jump procedure.

### 2.4.3 N-particle jump procedure

A jump procedure may be visualized as paths linking different configurations. But as mentioned in the above section, since the two-particle jump procedure preserves winding number, no path is built between two configurations

with different winding numbers. All the configurations are divided into groups according to their winding numbers; the two-particle jump procedure generates paths between configurations within the same group but no inter-group paths. If these were the only paths, the time average would be equal to ensemble average within one group, rather than over all configurations. So it is necessary to build paths joining different groups together.

Consider a simple case schematically illustrated in Fig. 5. The whole configuration set is divided into two groups. The paths between configurations within the same group are known but there is no path between two groups.

At first sight, we may think of building a path between the two groups. Let us choose configurations $S_1$ and $S_2$, and build a path between them; i.e. if the current state is $S_1$, then we may choose $S_2$ as a candidate for the next state. The probability of accepting $S_2$ as the next state is also given by the ratio of the probabilities of $S_1$ and $S_2$. This provides a chance of going between the two states of the groups. If the current state is $S_2$, also choose $S_1$ as a candidate of the next state. This scheme will connect the two groups by the path between $S_1$ and $S_2$.

But this will change the visit rate of $S_1$ and $S_2$. All configurations should be generated with equal probability, otherwise the visit rate of each configuration is not the same. This condition can be fulfilled if each of the configurations has the same number of configurations

directly connected to it. This can be simply demonstrated in Fig. 6.

In this example, there are three states and they are connected as shown in Fig. 6. Note that the state number 2 is connected to both state 1 and state 3, but state 1 and 3 are only connected to state number 2. If all the probabilities of hopping between states are one half, i.e. the probability of going to another state and stay in the old state are equal to one half; then using the Monte Carlo method to visit these three states, the visit ratio will be 1:2:1.

The visit frequency of a state depends on two factors. Note that a hopping process contains two steps; firstly, from all states connected to the current state, we randomly choose a state as the candidate of next state; then we determine whether to accept the new state as the next state or keep the old state as the next state by comparing the hopping rate and a random number. Let us consider the first step in more detail. If the current state is 1, we can only choose state 2 as the candidate of next state; if the current state is 2, we have two choices: either state 1 or state 3; so the probabilities of choosing states 1 and 3 are each only 50%. If the current state is 3, only state 2 can be chosen as the candidate of next state. We can say that the difference is due to the unsymmetrical connection between the states. From detailed balance, we can prove that the visit ratio is independent of the structure of the

connection but only on the number of connections of a state. The number of connections of a state is defined as how many other states are directly connected to that state. In this example, the number of connections of state 1 and 3 are 1; the number of connections of state 2 is 2. The actual visit ratio of two states is equal to the product of the ratio of the number of connections and the hopping ratio. For state 1 and 2, the hopping ratio is 1:1, and the number of connection ratio is 1:2, so the actual visit ratio is 1:2. By the same argument, the visit ratio of state 2 and 3 is 2:1. So, the visit ratio of the three states is 1:2:1.

In a real problem, the hopping ratio depends on the the Boltzmann factor $\exp(-\beta H)$, the number of connections depends on how you connect the states, or in other words, depends on the configuration generation procedure.

In the above example, if we want to have visit ratio 1:1:1 and keep the hop rates ( P(S) ) unchanged, we can add a path connecting state 1 to itself (also for state 3), then the visit ratio will be corrected to 1:1:1.

Let us construct a procedure for our fermion problem. As illustrated in Fig 7, if the fermions in i-1, i, and i+1 slices are alternately distributed on the sites, then by shifting the i slice left (right) by one lattice spacing, the winding number will increase (decrease) by one. For those configurations do not fulfil the above condition, we retain that configuration as the next state, or in other

words, force the configuration go back to itself through a virtual path.

Let us summarize the scheme as follows:

(1) Choose a time slice i.

(2) Check the fermion locations in slices i-1, i, and i+1.

(3) If the fermions are alternately distributed, perform (4)-(8); otherwise keep the current configuration as the next configuration.

(4) Randomly determine whether to increase or to decrease the winding number by one.

(5) Create a new configuration by shifting the slice i left (right) by one lattice spacing in order to increase (decrease) the winding number by one.

(6) Determine the hop rate R.

(7) Determine whether to hop or not by comparing the hop rate with a random number.

(8) If hop, update the checker-board configuration.

With the above procedure, we can generate a configuration with winding number n+1 (or n-1) from one with winding number n if the current configuration has three consecutive slices having fermions alternately distributed on the sites. If the current configuration does not have three consecutive slices having fermions alternately distributed on the sites, we retain the current configuration, just as a virtual path is built on that configuration. The above procedure is called N-particle jump procedure.

The advantage of this procedure is that we can calculate the probability of the new configuration easily. When using this procedure along with the two-particle jump procedure, it will not change the relative number of configurations connected to any configuration because the necessary virtual paths are automatically built.

Although the combination of the above two algorithms can ensure that we may visit all the allowed configurations, it will make the number of allowed configurations being sampled too large for our Monte Carlo method to be practicable. We do not have enough computer time even for the energy of the system to converge.

In order to reduce the number of configurations being sampled, we just sample configurations with winding number zero and one. The results are shown in the following chapter.

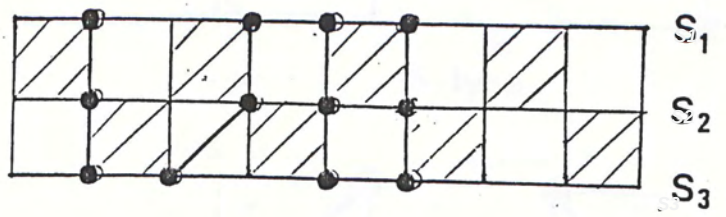



Fig. 1. Replace $\langle S_1|U_aU_b|S_3\rangle$ by $\Sigma\langle S_1|U_a|S_2\rangle\langle S_2|U_b|S_3\rangle$, summation over all allowed state $|S_2\rangle$

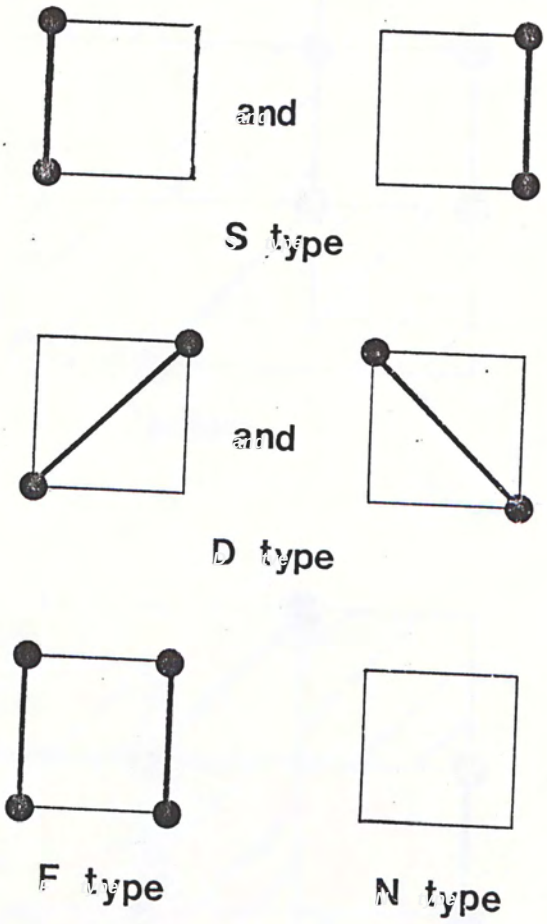
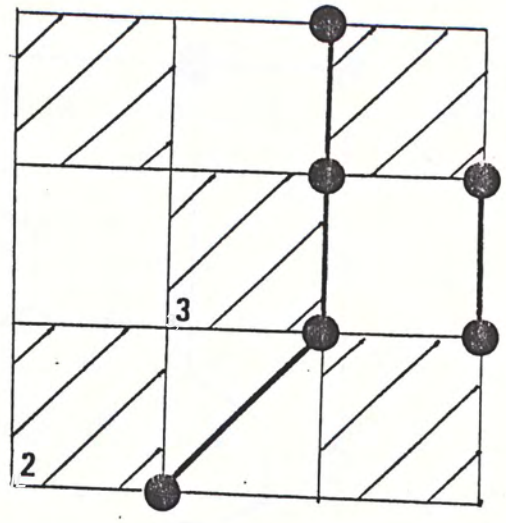



Fig. 2. Four allowed types of basic boxes.



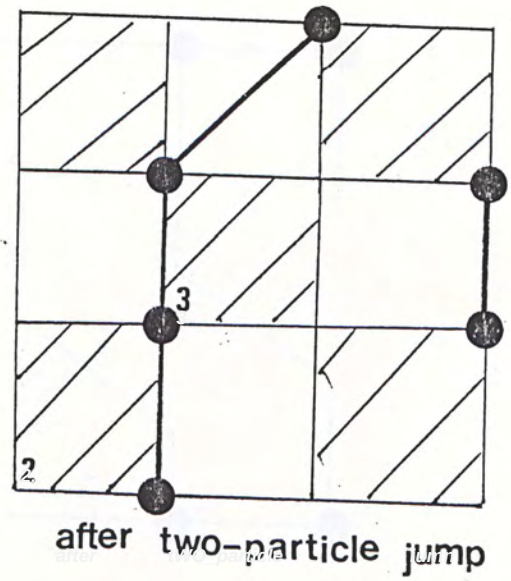
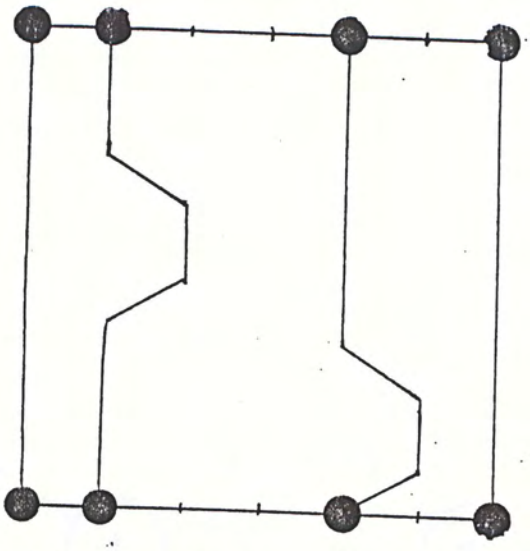
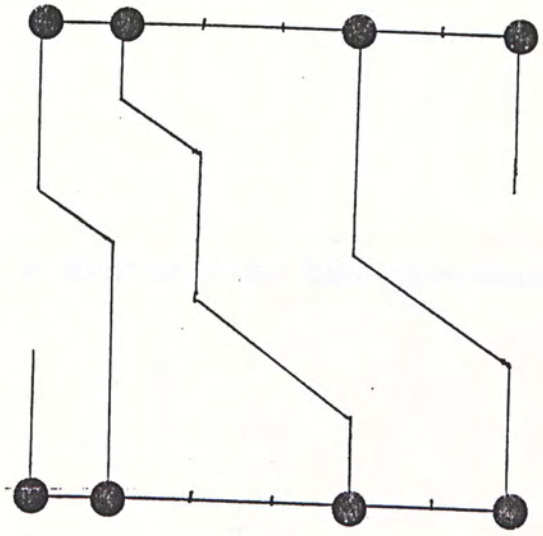




Fig. 3. Two-particle jump procedure



winding number zero configuration



winding number one configuration

Fig. 4. Winding number zero and winding number one configurations

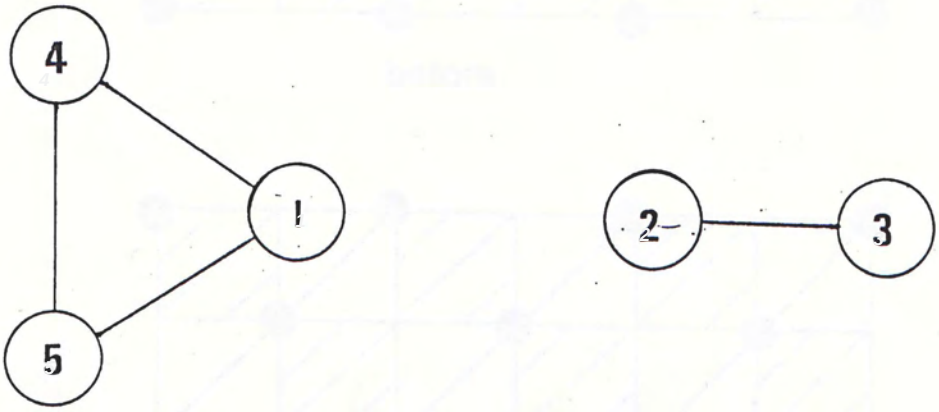



Fig. 5. A system with two disconnected classes

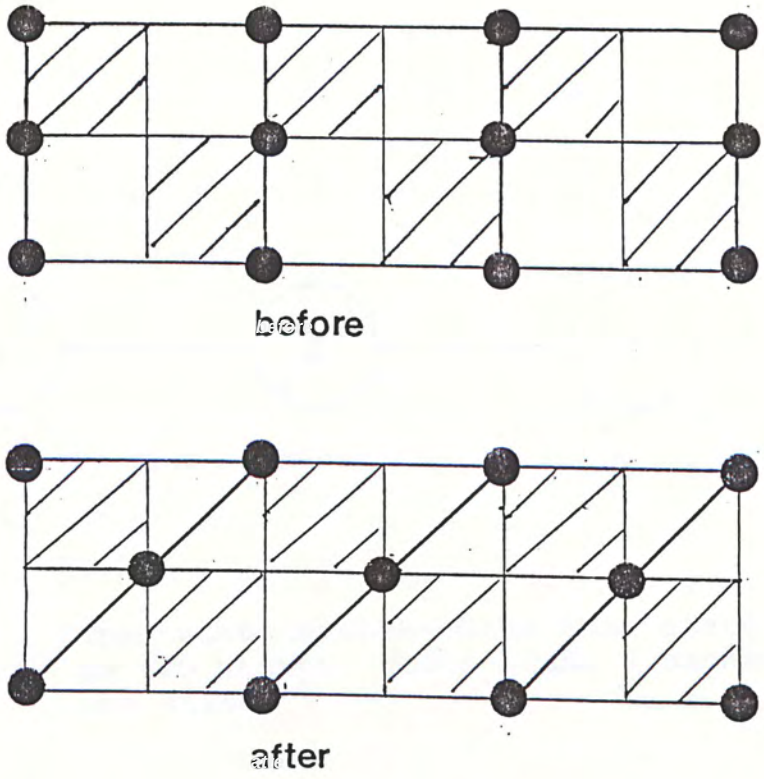



Fig. 6. N-particle jump procedure--the configurations before and after performing N-particle jump procedure are shown.

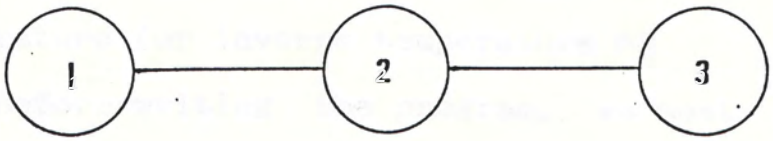



Fig. 7. Three-state system--Note that state 2 connects to two states, state 1 and 3 connect to only one state.

## CHAPTER 3

### Monte Carlo result

In this chapter, we will discuss the Monte Carlo program for the 1-D fermion system in more detail and present some results.

We shall calculate the internal energy and the specific heat per site of the fermion-system as functions of temperature (or inverse temperature $\beta$).

Before writing the program, we must choose a random number generator. What we must take care of is the period of the random number series generated. Since the number of the Monte Carlo sampling steps may be more than two millions in our problem, and three random numbers are needed in each step, the random number generator must have period larger than several millions. (More details can be be found in "Data Processing Technique", published by I.B.M.)

Before actual sampling, we usually perform $10^3$ pre-sampling Monte Carlo steps. The pre-sampling steps are the usual Monte Carlo steps, but the trace is not recorded for the averagings. Pre-sampling steps are used to reduce the initial 'transient' effect. In order to understand this, we must have the concept of probable region of the configurations. Recall that the probability of a configuration P(S) is a function of temperature and the numbers of different types of un-shaded boxes, Ns, Nd, Nf and Nn. Since we are using the two-particle jump procedure

to link up all the configurations, some regions of configurations may have larger probability for being visited than the others. Before simulation, we must choose a configuration as the initial state. If we are unlucky, we may choose a configuration with very small probability at that temperature. Although the detailed-balance condition will bring the trace back to the probable region, we may have to go through many configurations with small probabilities. In this case, if the trace is recorded for averaging, it means we will need more time to wait for the convergence. So in order to reduce the initial effect, we omit the first thousand steps as pre-sampling steps.

The results of internal energy and specific heat are shown in Fig. 8 and 9.

In our one-dimensional fermion problem, the hopping constant t is always set to 1. In Fig. 8 and 9, the nearest neighbor interaction constant v is zero, system length N is the number of sites. In Fig. 10 and 11, all parameters are unchanged except v is changed to 2.

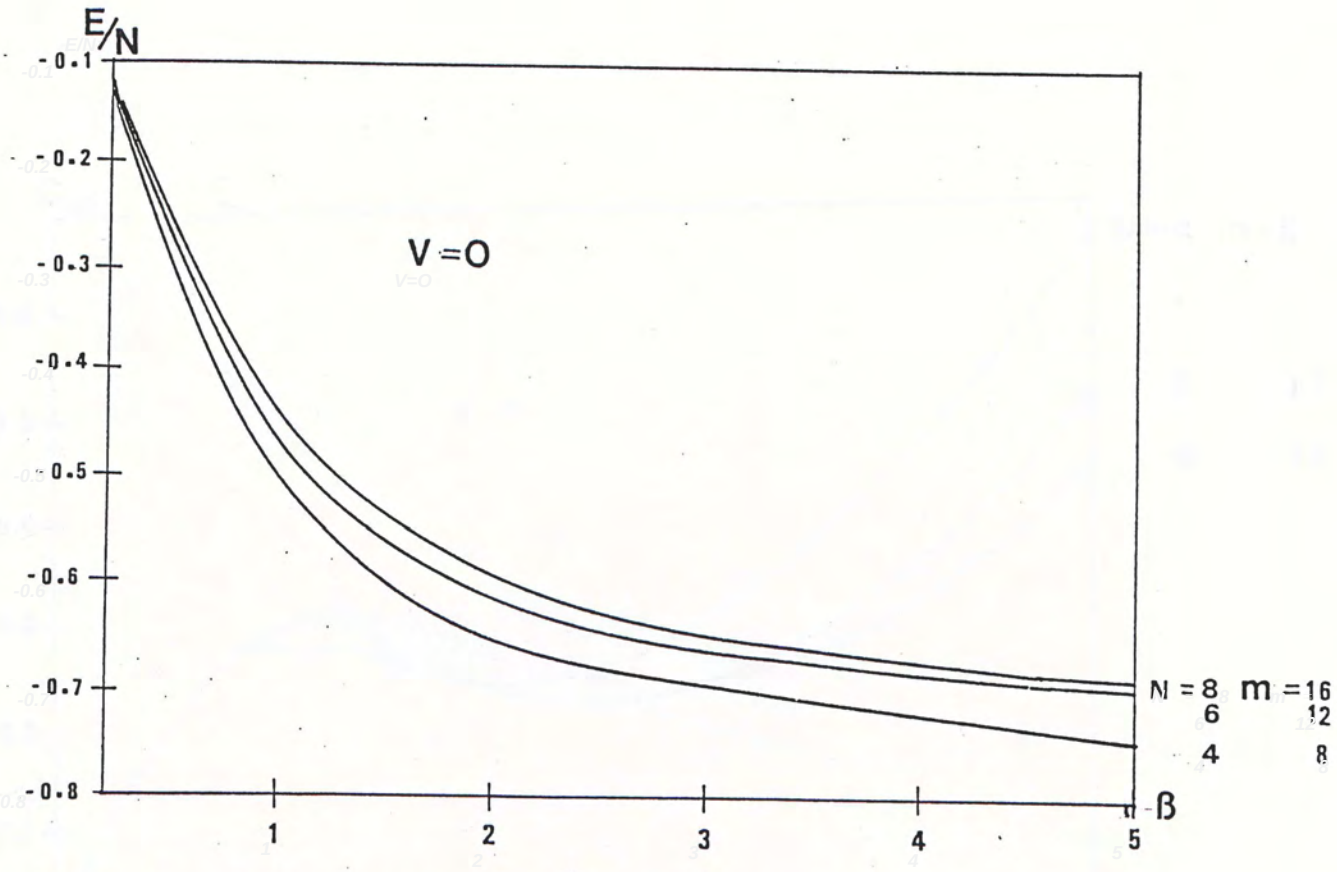



Fig. 8. Energy per site of 1-D fermion system. v=0.

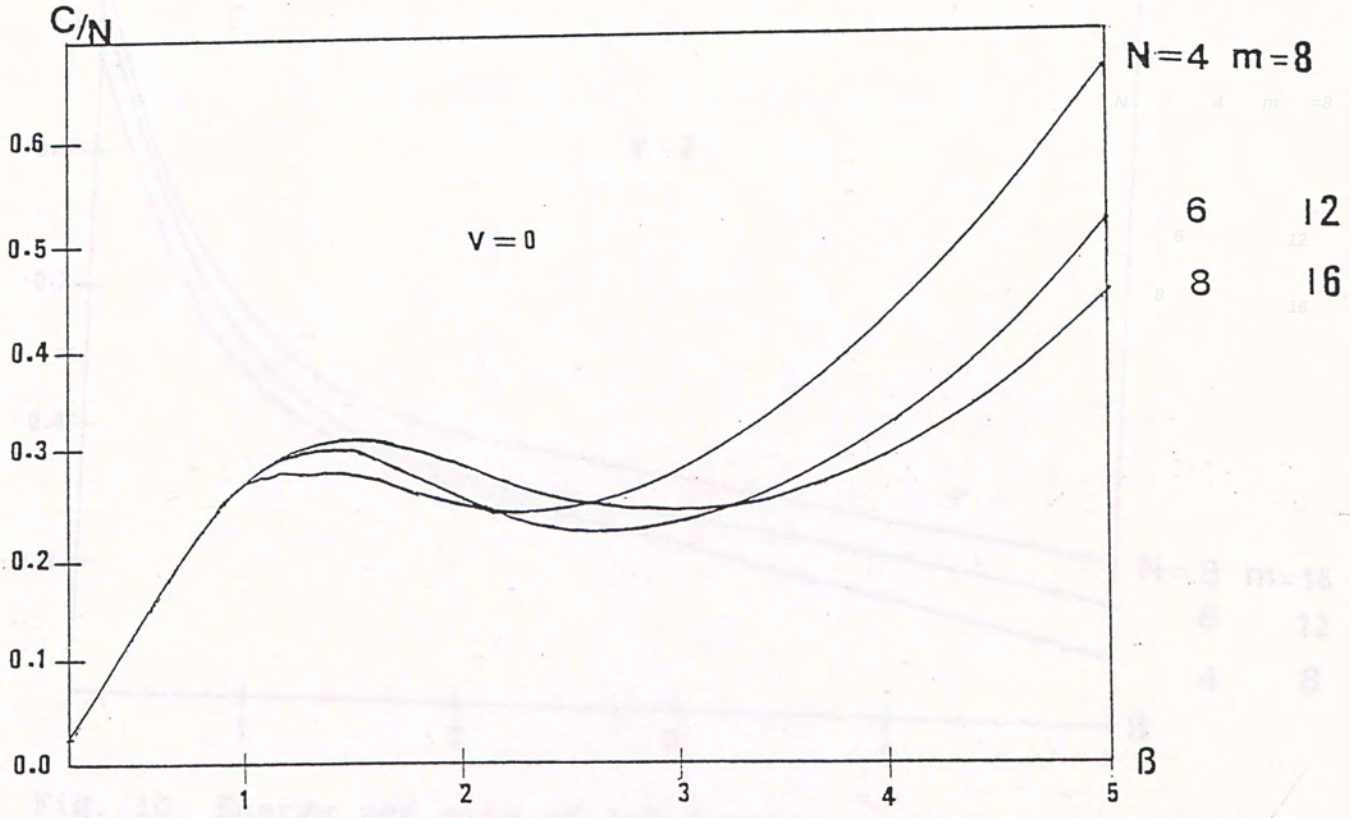



Fig. 9. Specific heat per site of 1-D fermion system.
v=0

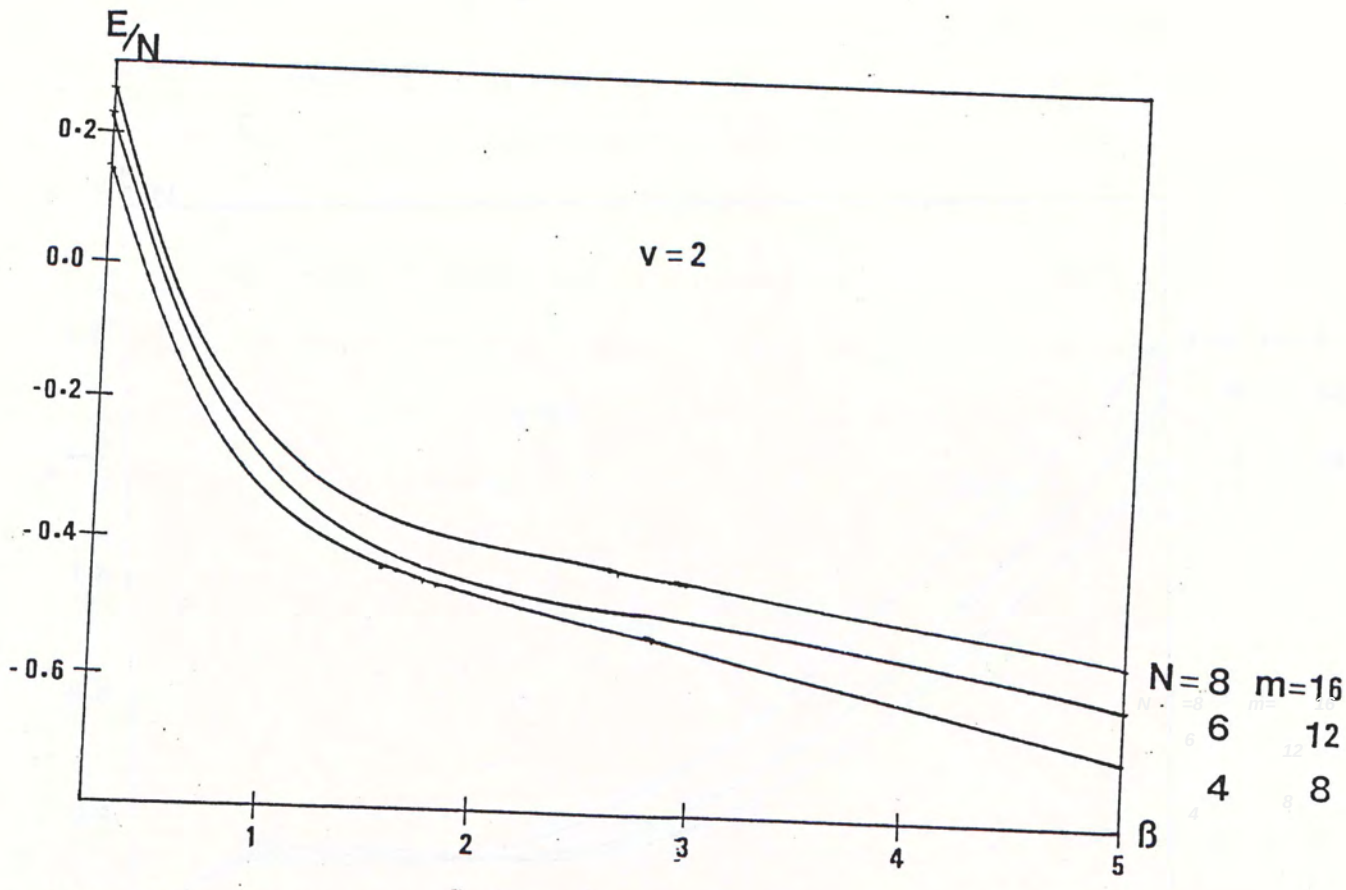



Fig. 10. Energy per site of 1-D fermion system. v=2

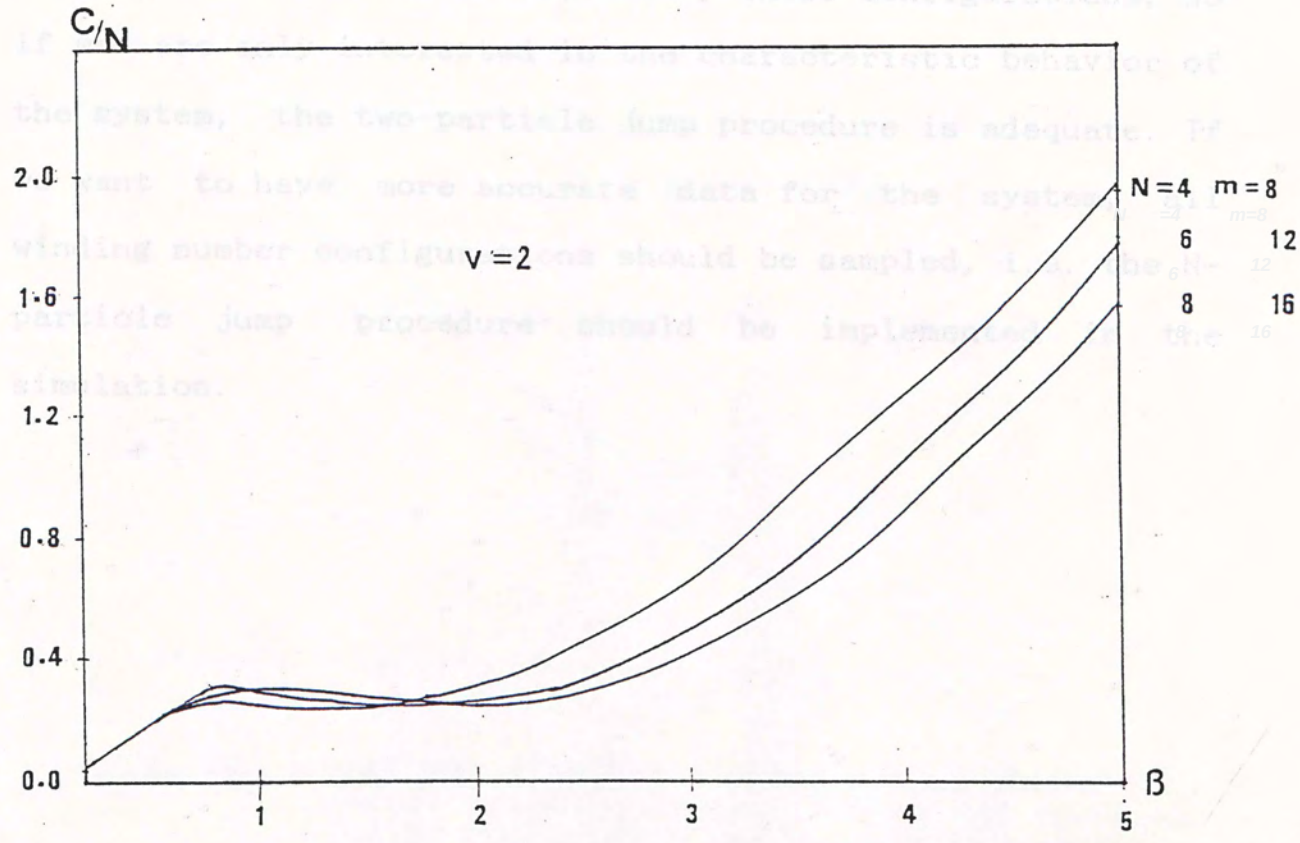



Fig. 11. Specific heat per site of 1-D fermion system. v=2.

As mentioned in the previous chapter, with the two-particle jump procedure, we can only sample the winding number zero configurations (if we start from a winding number zero configuration ), and the N-particle jump procedure has been introduced to handle this problem.

We have sampled the configurations with winding number zero and one to find the internal energy of the fermion system. The results are shown in Fig. 12 and 13.

From the results, we note that the winding number zero configurations are dominant states; the internal energy of the system is mainly determined by these configurations, so if we are only interested in the characteristic behavior of the system, the two-particle jump procedure is adequate. If we want to have more accurate data for the system, all winding number configurations should be sampled, i.e. the N-particle jump procedure should be implemented in the simulation.

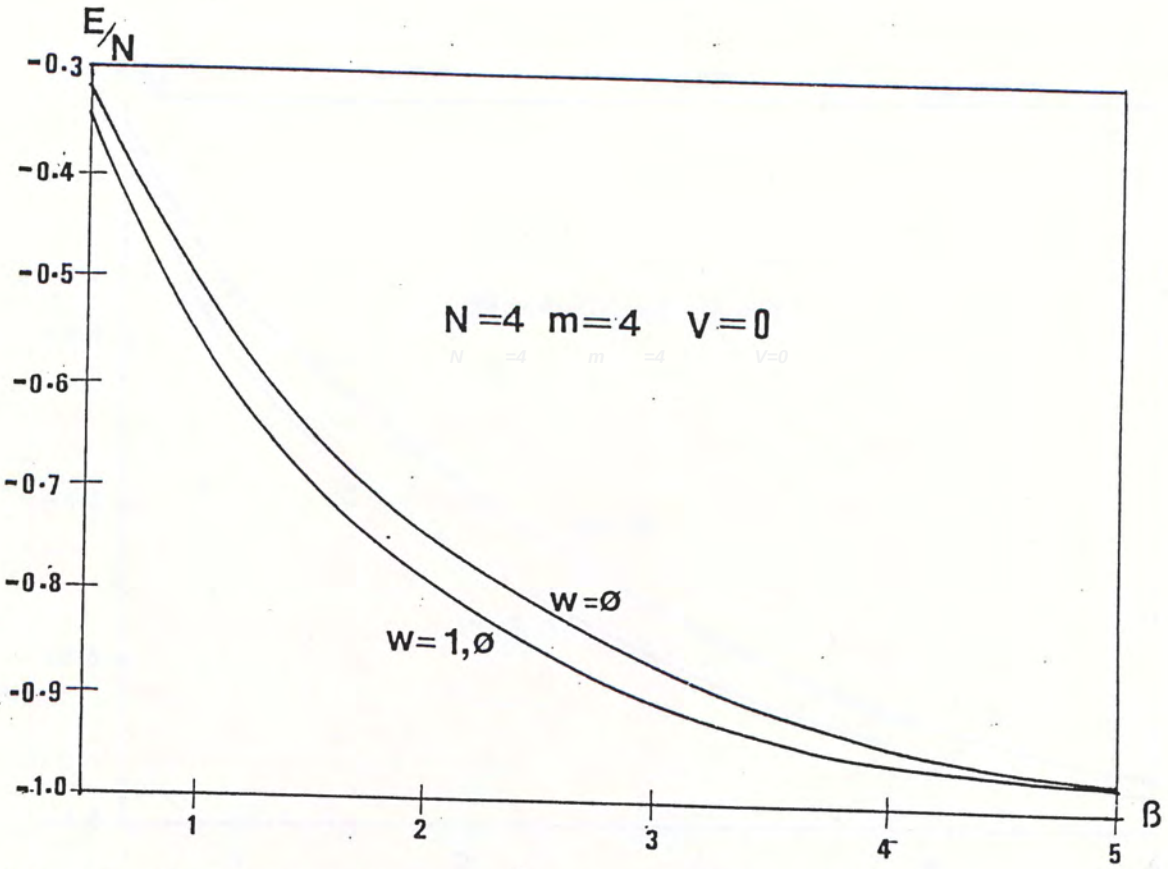



Fig. 12. Energy per site for winding number zero configurations only (w=0 curve) and for both winding number zero and one configurations (w=1,0 curve)

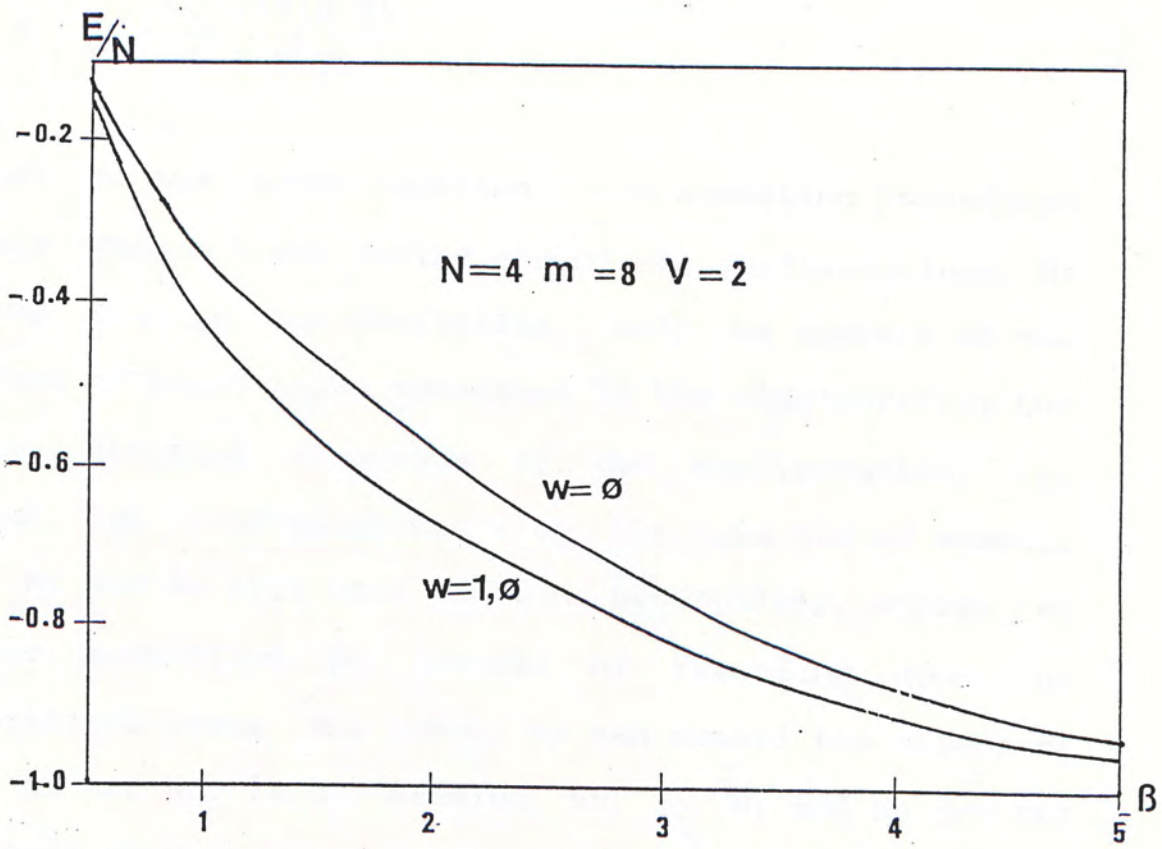



Fig. 13. Specific heat per site for winding number zero configurations only and for both winding number zero and one configurations

# CHAPTER 4

## modified Monte Carlo method

### 4.1 Degeneracy method

In our one-dimensional fermion problem, we are taking averages over all the configurations by the Monte Carlo sampling method. The basic formula is the following equation:

$$< E >_{mc} = \frac{\sum P(S)E(S)}{\sum P(S)}$$

Note that in the above equation, the summation is defined along any random trace moving around the configurations. We also note that in our derivation, only the numbers of the four types of basic boxes contained in the configuration but not the detailed structure of the configuration, are essential. Two configurations with the same set of numbers Ns, Nd, Nf and Nn will have the same probability, energy and all other quantities. So instead of recording down the configurations along the trace, we can record the values of Ns, Nd, Nf and Nn. In our problem, Ns, Nd, Nf and Nn are not independent numbers; they fulfil the following equations:

$$Ns + Nd + Nf + Nn = N * m$$

and

$$Nf = Nn$$

where N is the site number of the system, 2m is the length of the Trotter $\tau$-dimension. The second equation can be derived from the conservation of total number of fermions in the checker-board.

From the above equations, instead of recording four numbers (Ns, Nd, Nf and Nn) for a configuration, we can record any two of the four numbers, we choose Ns and Nd. So from now on, we use a set of two numbers Ns and Nd to denote a configuration.

Note that the mapping between the configurations and their Ns and Nd numbers is a many to one mapping, two (or more) different configurations may have the same set numbers of Ns and Nd. This allows us to treat the Monte Carlo averaging from another point of view.

In general, configurations can be classified according to their probabilities P(S), i.e. all configurations having the same probability P(S) are grouped to form a class. If we forget about the trace operator by means of the Monte Carlo sampling, but only record down how many times a configuration has been visited ( or frequency F(S) ), we will find that the number of a configuration being visited is proportional to the probability of the configuration P(S). Now, we define the frequency F(Ns,Nd) in the following manner:

$$F(Ns,Nd) = \Sigma' F(S)$$

where the summation is over all states having Ns type-s and Nd type-d un-shaded boxes.

We have the following equations:

$$F(S) = \frac{P(S)}{\Sigma P(S)} * \text{total number of sample steps}$$

and

$$F(Ns,Nd) = \frac{g(Ns,Nd)\ P(Ns,Nd)}{\Sigma\ g(Ns,Nd)\ P(Ns,Nd)} * \text{total number of sample steps}$$

the g(Ns,Nd) is defined as the number of configurations having Ns type-s and Nd type-d un-shaded boxes. We will call this factor as degeneracy.

The probability of a configuration ( P(S) ) depends on the numbers Ns and Nd of the configuration and the temperature (or inverse temperature β). But the degeneracy ( g(Ns,Nd) ) is independent of temperature. (In some cases, we write P(S;β) as P(S), one should note that they are the same function). With the introduction of the degeneracy factors g(Ns,Nd), the temperature-dependent Monte Carlo sampling can be modified to be temperature-independent.

We may start from rewriting the following equation:

$$< E >_{mc} = \frac{\sum P(S;\beta)\ E(S;\beta)}{\sum P(S;\beta)}$$

where the summation is along a random Monte Carlo path at temperature $\beta$, to

$$< E >_{mcd} = \frac{\sum g(Ns,Nd)\ P(Ns,Nd;\beta)\ E(Ns,Nd;\beta)}{\sum g(Ns,Nd)\ P(Ns,Nd;\beta)} \qquad (4.1)$$

where the summation is over all allowed values of Ns and Nd. The $<..>_{mcd}$ is used to denote the modified Monte Carlo degeneracy method. The $\beta$ has been put in the equation in order to show the dependence.

In equation (4.1), the degeneracy factor g(Ns,Nd) is unknown while the P(Ns,Nd;$\beta$) and E(Ns,Nd;$\beta$) are all known functions. Note that g(Ns,Nd) is the number of ways in which Ns type-s and Nd type-d un-shaded boxes can be put together to build an allowed checker-board configuration. It is a very complex function of Ns and Nd, we do not know how to solve it analytically. But in order to perform averaging, only the relative ratio of g(Ns,Nd)'s is meaningful; so the basic Monte Carlo method may help.

If we use the basic Monte Carlo method to travel along the configurations, and set all the probabilities P(S) to 1 ( this may be called a zero-$\beta$ sampling since P(S)~exp(-$\beta$E), as $\beta$->0, P(S)->1), then the number of visits to the

configurations having Ns type-s and Nd type-d un-shaded boxes is proportional to g(Ns,Nd).

Let us summarize the ideas. Firstly, we perform a Monte Carlo simulation with all P(S)'s equal to 1; this will give us a set of relative ratios of g(Ns,Nd)'s. Secondly, with the relative ratio of g(Ns,Nd) and the function forms of P(Ns,Nd;β) and E(Ns,Nd;β), we perform averaging at different (inverse) temperature β according to equation 4.1.

The advantage of the modification mentioned above is that we are performing sampling over all allowed configurations to find a temperature independent function g(Ns,Nd), but not performing sampling at a fixed temperature to find E(β); we can calculate the thermal properties for all temperatures by only performing one zero beta sampling in order to get the relative degeneracy factors.

## 4.2 Example

In this section, we will apply the modified Monte Carlo method to the following two-dimensional system. The computer program was developed by K.Y. Chan.

Consider a particle moving freely within a two dimensional 11*11 region. An allowed state of the particle S(i,j) is characterized by its x-y coordinates i and j, where the indices i and j go from zero to ten.

Given that the energy of the state S(i,j) is i+j, i.e. E(i,j)=i+j, then the probability of finding the particle in

state S(i,j) is proportional to exp[-β(i+j)]. The degeneracies are as follows:

For n=i+j,

$g(n) = n+1$ $\quad 0<=n<=10$

$g(n) = 20-n+1$ $\quad 20>=n>=10$

and we have

$$Z = \sum_{i,j} \exp[-\beta(i+j)] \qquad i,j=0,1,2,\ldots 10$$
$$= \sum_{n} g(n)\, P(n)$$
$$= \sum_{n} g(n)\, \exp(-\beta n)$$

summation n is from 0 to 20 (the lowest to highest energy state).

For this system, the thermal properties such as the internal energy can be solved exactly.

The results for the specific heat calculated by Monte Carlo and the modified Monte Carlo method are shown in Fig. 14. 50000 Monte Carlo steps have been performed for each temperature. The modified method used the degeneracy factors obtained from the zero beta data, i.e. only one simulation at zero beta gives result for all temperatures. As we can see from Fig. 14, the error is large near the peak of the specific heat (per site C/N). This is due to the fact that at temperatures far from the maximum point, the particle will stay only at very high or very low energy states, so it is much easier to get the correct result from the Monte Carlo sampling. In Fig. 15, 125000 Monte Carlo steps have

been performed. We find that the accuracy of the modified method result was improved. But the result of the traditional Monte Carlo method showed no improvement.

It is very easy to find the internal energy by both methods. The results are found to be virtually exact.

The modified method was rather successful and reduced a lot of the Monte Carlo steps in solving this problem. It should be noted that this success resulted from the fact that the system under consideration is a very small one.

In general, at any temperature, some configurations have much larger visit probabilities than the others. Even if we do not include configurations with low probabilities in the sampling, we may still have a rather good result. In the old Monte Carlo simulation, the probability factor $P(S;\beta)$ automatically bring the sampling trace around the large probability configurations, so the number of Monte Carlo steps needed to get acceptable data is proportional to the size of the most probable region at that temperature. But in the modified Monte Carlo method, we need to find the degeneracy factor g for all classes. Note that g appears with $P(Ns,Nd)*\exp[-\beta E(Ns,Nd)]$ in our derivations, so even the small g classes may be important at some temperatures. In order to find g(Ns,Nd) by zero-beta Monte Carlo sampling, the number of steps is proportional to the number of all allowed configurations. This is usually a much larger number than the number of configurations in the most probable region at any temperature.

If we know the analytic form of the degeneracy $g(n)$, we have no trouble; if we do not know its form, whether the modified method can give benefits depends on how smooth the degeneracy $g(n)$ curve is. For a smooth curve, the Monte Carlo method is a very effective tool since a relative small number of simulations may give the accurate shape of the curve.

Note that in the above example, the maximum $g(n)$ is 11 and the minimum $g(n)$ is 1, the difference is just one order. But in some problems, the difference may be up to 10 orders. Consider a group of non-interacting defects in a 10*10 crystal lattice. The number of defects is not fixed but only one defect may appear at a given site at any time. The energy of the system is given by the number of defect in the system n, i.e. $E(n)=n$. Then the minimum $g(n)$ is 1 (no defect in the system), but the maximum $g(n)$ is about 100!/50!*50!, their difference is more than one hundred orders. For such cases, the simple modified Monte Carlo method discussed above may not give acceptable results since the Monte Carlo program will simulate only the configurations with large $g(n)$. The number of steps needed to get an accurate $g(n)$ depends on the ratio of maximum and minimum $g(n)$'s. (This is also the problem of the old Monte Carlo method that only the large $P(S)$ configurations are sampled.)

In order to solve this problem, We make another modification as discussed in the next section.

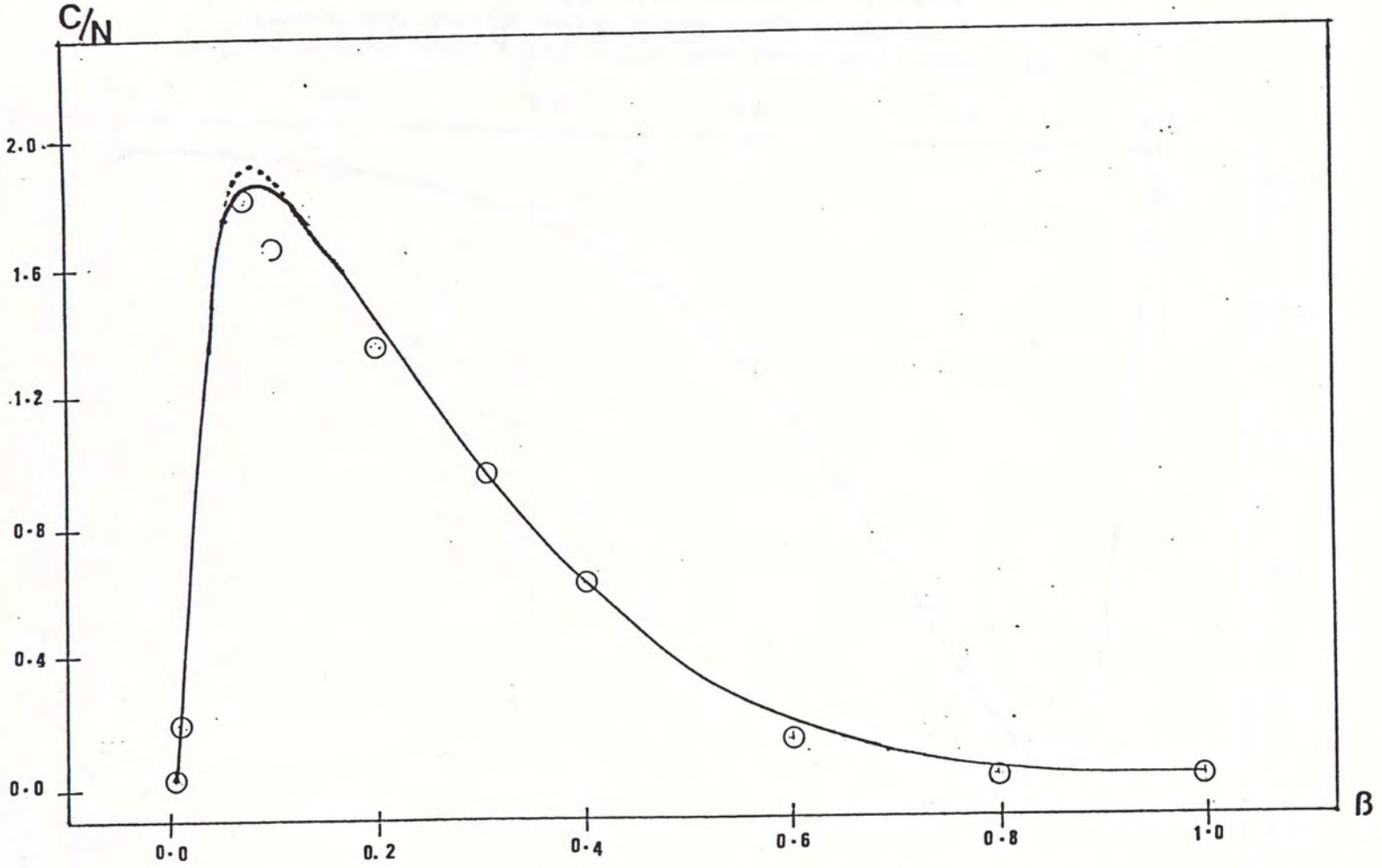



Fig. 14. Specific heat per site from degeneracy method-50000 steps. Solid line gives the exact result, circles give the Mc result, dashed line gives the degeneracy MC result based on the MC result at $\beta=0$.

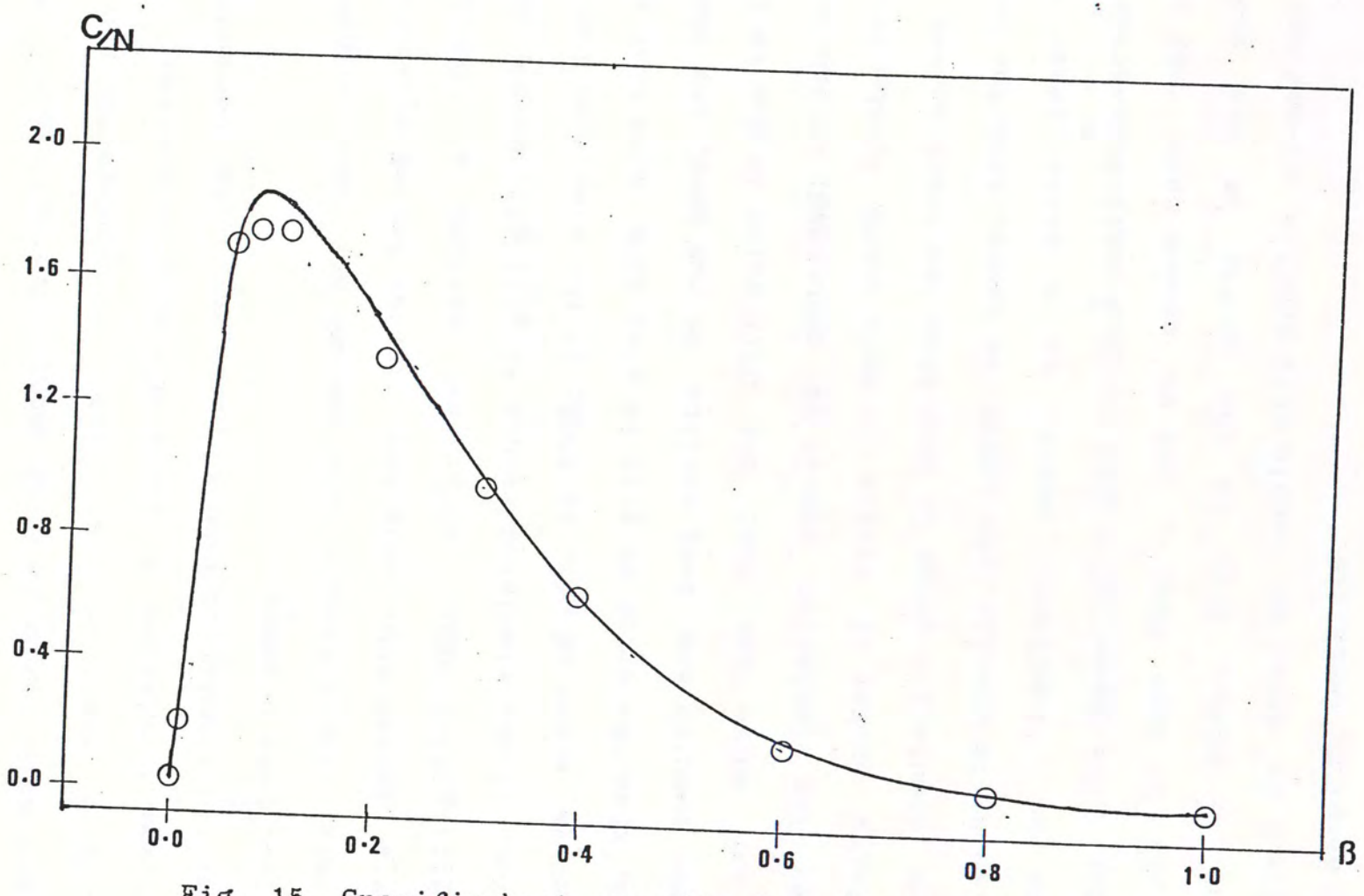



Fig. 15. Specific heat per site from degeneracy method- 125000 steps. Solid line gives the exact result, circles give the Mc result, the degeneracy MC result based on the MC result at $\beta=0$ virtually coincides with the exact result.

## 4.3 Weighted degeneracy Monte Carlo method

From the above discussion, we want to get the relative g(n)'s by Monte Carlo sampling among different classes. The difficulty is that for some classes the degeneracy g(n)'s are too small relative to those of the other classes; this will make the simulation results carry very large percentage errors for these classes.

A very simple example may demonstrate this problem. Consider a system with only two classes having degeneracy ratio g(1):g(2)=1:1000. After we perform a 500-step simulation, if the simulated result is g(1):g(2)=0:500, then the relative error of g(1) is 100%; if the simulated result is 1:499, then the error of g(1) is also 100%. Note that the above two results are best results we can have, any other results are worse than that. The difficulty is due to the fact that the simulation result is quantized; we can only have integer number of visits for each state. If the total number of simulation steps is less than the total number of states (this is usually the cases we have), then the non-existence of 'factional' visit to a state leads to difficulty. (The number of visits to some small degeneracy g classes may be less than 1, but we cannot have, say 1/2 visit of a state, this is the source of the above difficulty.) In order to handle this problem, we may try a weighted sampling technique.

Consider the above two-class system again. (recall that g(1):g(2)=1:1000) The g(n)'s given by the Monte Carlo simulation at zero beta, (i.e. with equal unit probabilities, P(1)=P(2)=1 ) are the numbers of visits V(n) to each states, nearly proportional to degeneracy g(n). If we change P(1) to 1000 and keep P(2) unchanged, and then perform the Monte Carlo simulation for 500 steps, the numbers of visits to the two states should be nearly equal to 250. Assume we have 10% simulation error, i.e. V(1) is 225 and V(2) is 275, then after rescaling the results, we get g(1):g(2) = 225/1000:275/1 ≈ 1:1222; so the results we get carry about 20% error, it is much better than the 100% error we have in the last treatment.

In general, the probability we would like to assign to each class is 1/g(n) such that we have the same number of visits to each class. Although we do not know the values of g(n)'s, we can make use of following iteration scheme:

(1) Guess a set of g(n)'s according to the problem. (If you know nothing about the g(n), set them to unity.)

(2) Set P(n) as 1/g(n).

(3) Perform the Monte Carlo simulation using the probabilities P(n)'s for N times.

(4) If the number of visit to each class found in the last step is V(n), then reset the probability P(n) as P(n)/V(n) for each configuration.

(5) Perform another N-step Monte Carlo simulation.

(6) Repeat steps (4) and (5).

(If some V(n)'s found in step 4 are zero, try to increase the P(n)'s for those configurations and/or the number of simulation step N. If these changes do not help, it means that the g(n)'s of those configurations are too small to be sampled relative to the g(n)'s of the others)

By the above iteration method, the relative number of visit of each class found in the last step will be more likely to have the same order of magnitudes. The degeneracy is given by V(n)/P(n).

Another method which may be used to solve the problem is to perform some simulations with different weighting factors and combine the results to find the whole set of g(n)'s. Consider another three-class system as an example. The degeneracy are 1, $10^3$ and $10^6$ respectively. If the resolution of the sampling is about 1/1000, then we can only find g(2) and g(3) but not g(1). In this case, we can perform two separate simulations, one with P(1)=0, P(2)=P(3)=1, another with P(1)=P(2)=1 and P(3)=0. The first sampling will give $g(2):g(3)\approx 1:1000$, the second simulation will give $g(1):g(2)\approx 1:1000$. After combining the two results, we get $g(1):g(2):g(3)\approx 1:10^3:10^6$.

In general, there are many classes in the system, and the g's are not yet known, so it is not possible to assign correct weighting factor for each class. What we actually do is to make use of the Boltzmann factor $\exp(-\beta E)$. We perform the Monte Carlo simulation at different temperatures to find some sets of g(n). Each of these g(n)'s will not be accurate

for all of the allowed n but it will have high accuracy around some $n^*$ which is controlled by $\beta$. By varying the Boltzmann factor, we shift $n^*$.

The crucial step is to devise a scheme to combine the various sets of data into a single reliable $g(n)$ with which we can calculate the internal energy and other thermal properties.

# Chapter 5

## Small beta expansion

All of our derivations started from the Trotter expansion of the partition function. The error is proportional to $\beta/m$, where $m$ is the length of the Trotter dimension, so all the Monte Carlo results carry larger error at low temperatures. In order to get accurate result at larger beta, the direct way is to use large $m$, another way is to extrapolate from small beta results.

Increasing $m$ will make the checker-board large, i.e. increase the number of allowed configurations in the sampling space. The increase of the number of allowed configurations explodes so fast that we might not be able to afford the computer time for the Monte Carlo sampling. In the following, We will discuss the small beta extrapolation based on the Padé approximant.

For our one dimensional fermion problem, the internal energy should approach to the ground state energy at large beta, this suggests a [L]/[L] Padé approximant is favorable.

### 5.1 Padé approximant

Recalling the expression for the internal energy, for a N-site system with the length of the Trotter dimension equal to m (i.e. the size of the checker-board is N*m), we have

$$E = \langle E \rangle_{mcd}$$

$$= \frac{\sum g(Ns,Nd)\ P(Ns,Nd)\ E(Ns,Nd)}{\sum g(Ns,Nd)\ P(Ns,Nd)}$$

The summation is over all possible Ns and Nd. And

$$P(Ns,Nd) = Ps^{Ns} * Pd^{Nd} * Pf^{Nf} * Pn^{Nn}$$

$$E(Ns,Nd) = Es*Ns + Ed*Nd + Ef*Nf + En*Nn$$

and

$$Nf = Nn = ( N*m - Ns - Nd )/2$$

where

$$Ps = \exp(\beta V/m)\ \mathrm{Cosh}(2\beta t/m)$$

$$Pd = \exp(\beta V/m)\ \mathrm{Sinh}(2\beta t/m)$$

$$Pf = \exp(-\beta V/m)$$

$$Pn = \exp(-\beta V/m)$$

$$Es = -\frac{V}{m} - \frac{2t}{m}\ \mathrm{Tanh}\left(\frac{2\beta t}{m}\right)$$

$$Ed = -\frac{V}{m} - \frac{2t}{m}\ \mathrm{Coth}\left(\frac{2\beta t}{m}\right)$$

$$Ef = \frac{V}{m}$$

$$En = \frac{V}{m}$$

Replacing Nf and Nn in terms of Ns and Nd, we have

$$P(Ns,Nd;\beta) = \exp(-N\beta v/2) * \exp[-\beta v*(Ns+Nd)/m] * [\mathrm{Cosh}(2\beta t/m)]^{Ns} * [\mathrm{Sinh}(2\beta t/m)]^{Nd}$$

Note that $\exp(-N\beta v/2)$ is a constant term, so $P(Ns,Nd;\beta)$ can be rewritten as follows:

$$P(Ns,Nd;\beta) = \exp[-\beta v*(Ns+Nd)/m] *[\cosh(2\beta t/m)]^{Ns}*[\sinh(2\beta t/m)]^{Nd}$$

$$E(Ns,Nd;\beta) = -(Ns+Nd)*2v/m + Nv/2 - [Ns*\tanh(2\beta t/m)+Nd*\coth(2\beta t/m)]*2t/m$$

$$E = \frac{\sum g(Ns,Nd)\ P(Ns,Nd)\ E(Ns,Nd)}{\sum g(Ns,Nd)\ P(Ns,Nd)}$$

The denominator and numerator in the above equation are now expanded separately to two L-1 order polynomials of $\sinh(2\beta t/m)$. In order to keep the leading terms, we only need to have the $g(Ns,Nd)$ for small Nd since the terms are in the form of $g(Ns,Nd)*\sinh(2\beta t/m)^{Nd}$. Therefore we perform simulation to find $g(Ns,Nd)$ for small Nd class. As the approximated formula is in the [L]/[L] form, it guarantees that at large beta, the energy will tend to a constant value.

In Fig. 16, we show that the exact internal energy per site for a non-interacting system (i.e. v=0) of two fermions on four sites. The result based on the [3]/[3] approximant (i.e. we keep up to $\sinh(2\beta t/m)^2$ both in the denominator and numerator) for a system with N*m=4*8 is also presented. We observe a close agreement between the results up to $\beta$=2, which lends credence to the validity of our method of extrapolation. For the sake of comparison, results for systems with dimensions 4*6 and 4*4 are also shown. As expected, the accuracy improves with increasing m.

Figure 17 shows the internal energy, again based on the [3]/[3] approximant, for interacting systems with v=2 but with different lattice size N.

For N=8 and m=16, our (E/N) has the value -0.345 at $\beta$=2 and -.357 at $\beta$=4. These values compare favorably with the exact results -0.360 and -0.369 at these two temperatures. (Ref. H. De Radet and Ad Langendijk, 1985, Phys. Rpts. 127,233)

Figure 18 shows the internal energy for interacting systems with v=2 and N=8 but with different length of the Trotter dimension m. Figure 19 shows the internal energy for interacting systems with v=2 and N=6 also with different length of the Trotter dimension m. The m dependence is found to be weak at very low and very high temperature.

In view of the above, we expect that the small Ns simulation in conjunction with the Padé approximant can provide useful estimates for thermodynamic quantities at relatively low temperatures.

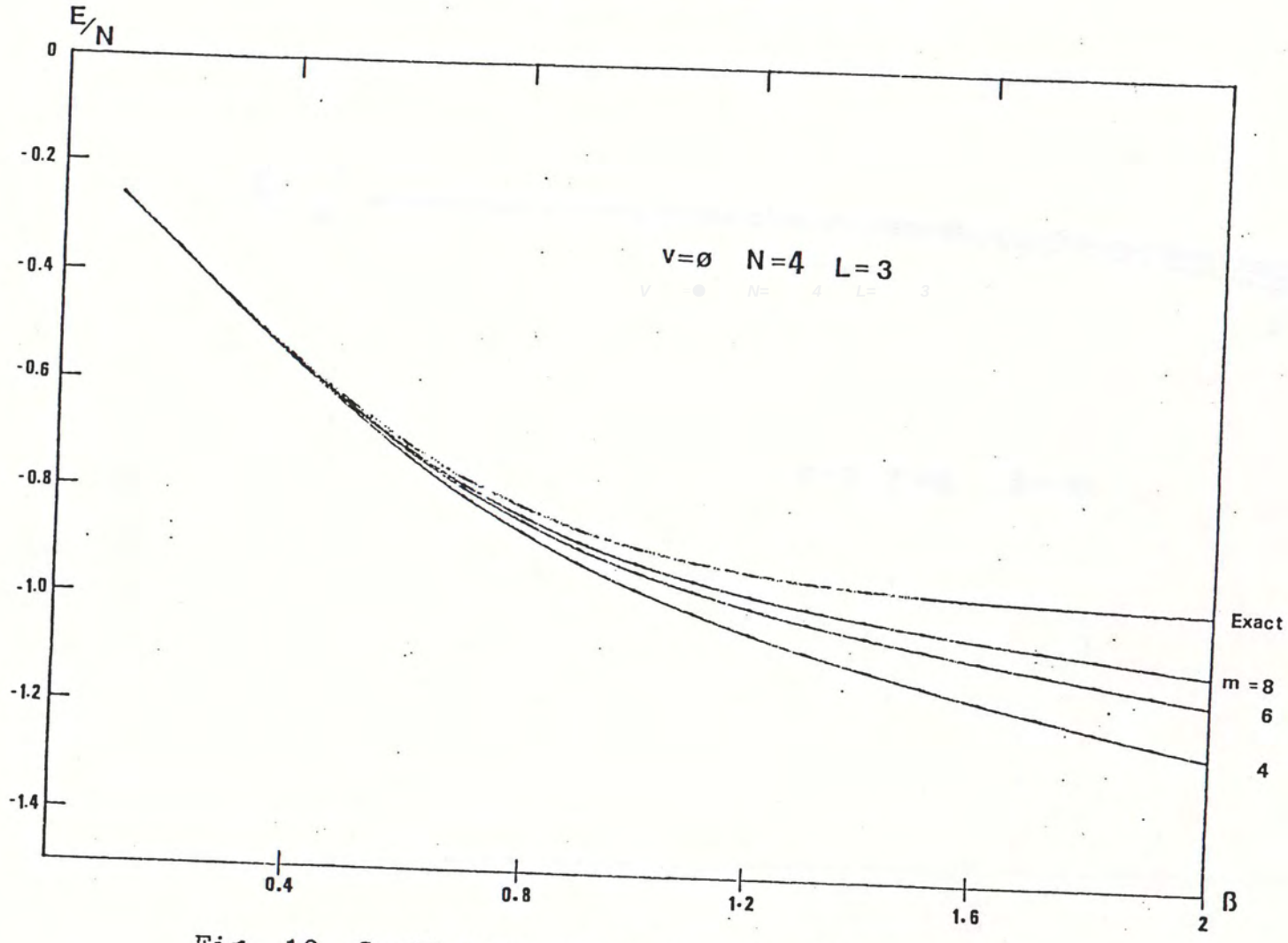



Fig. 16. Small beta expansion for E/N--v=0, N=4

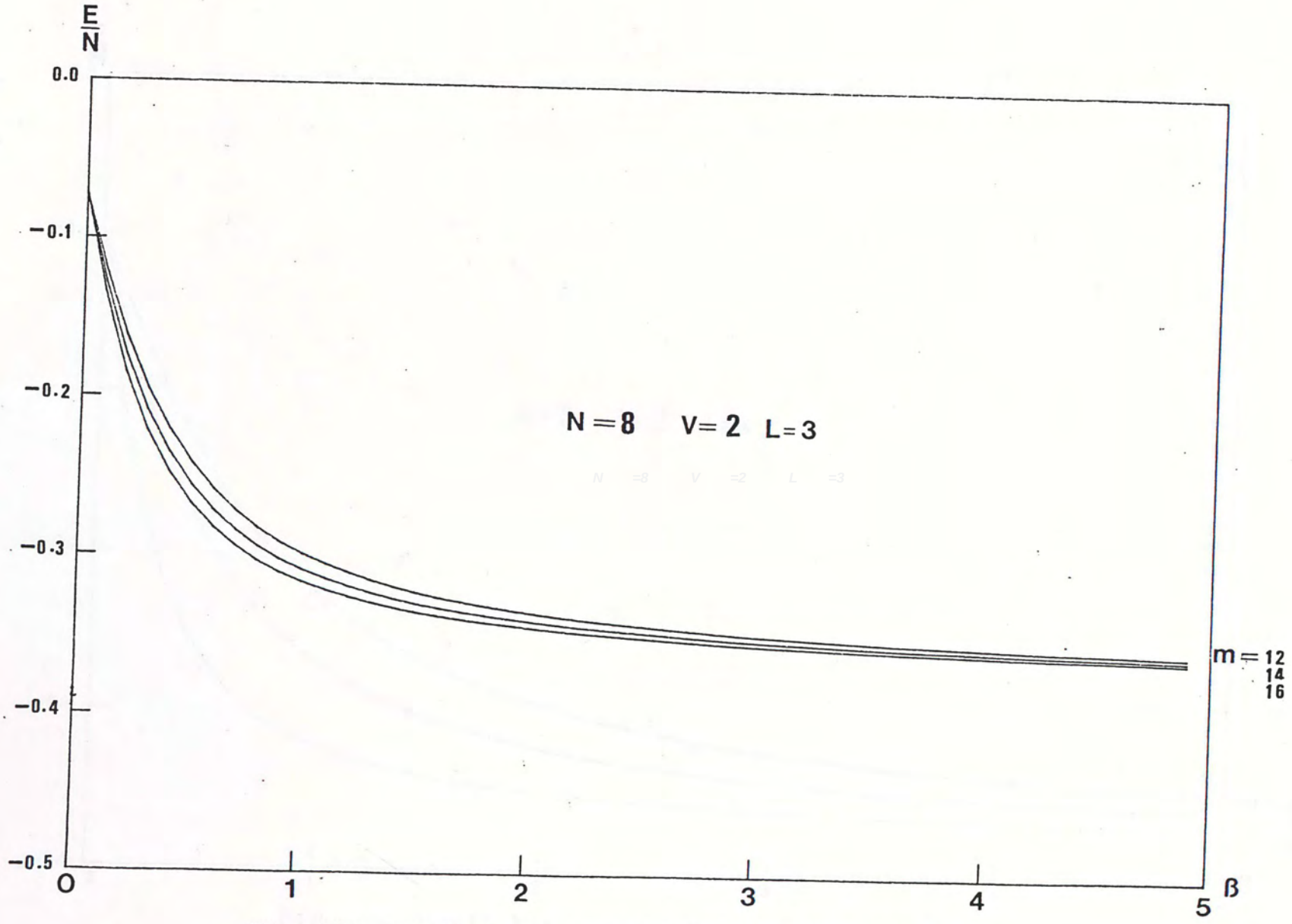



Fig. 17. Small beta expansion for E/N--v=2, N=8

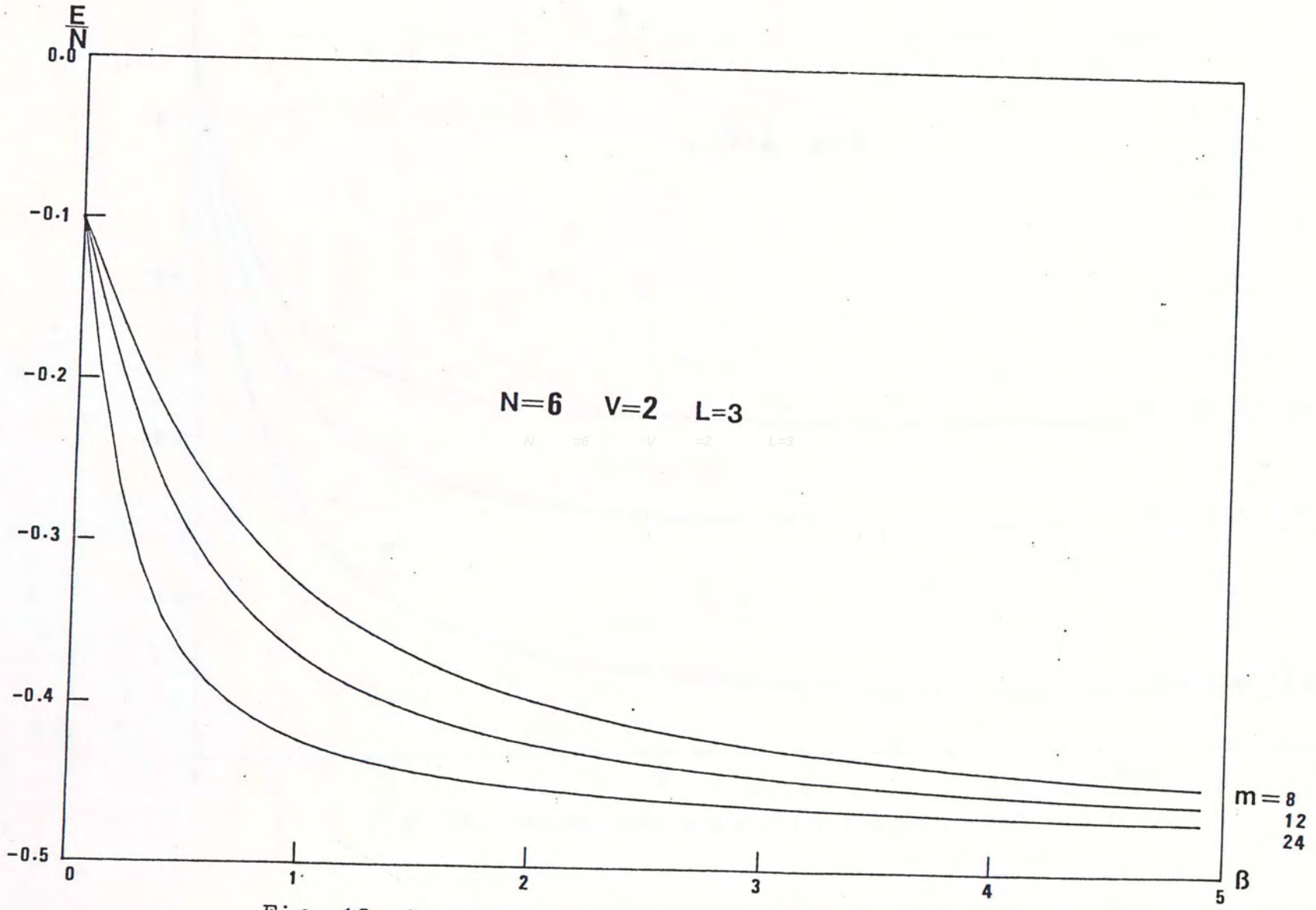



Fig. 18. Small beta expansion for E/N--v=2,N=6

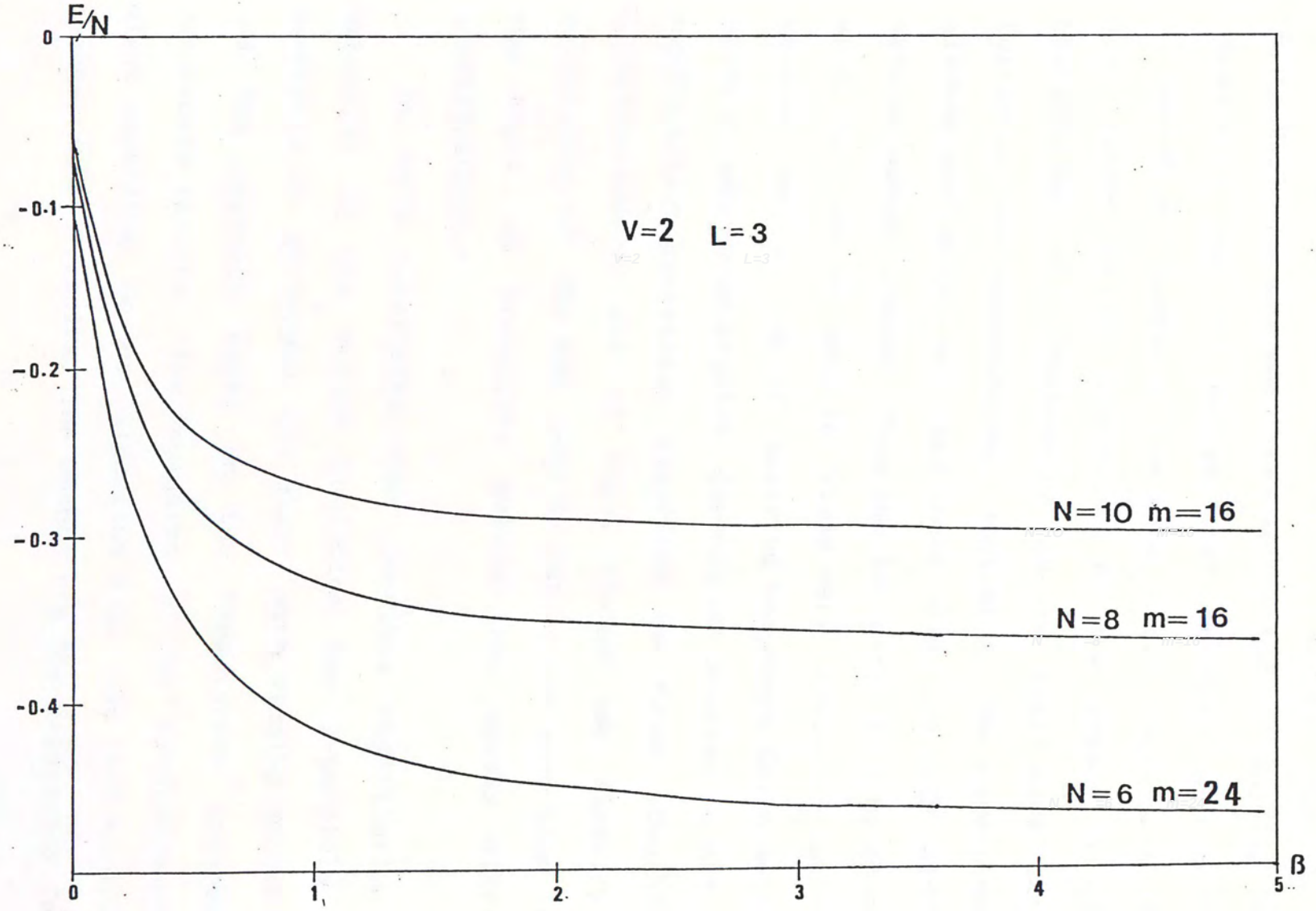



Fig. 19. Small beta expansion for E/N--v=2,N=6,8,10

# Chapter 6

## Discussion and conclusion

In the pervious chapters, we have discussed the basic Monte Carlo Method and its application in solving the 1-D fermion problem. We realize that the Monte Carlo method as discussed in chapter 2 has some basic limitations. One is the Trotter error introduced in the expansion of the Hamiltonian, which forbids us to use small size checker-boards at low temperatures. Another is the huge number of allowed configurations, and their division into different winding number classes. Thus the two-particle jump procedure alone may not be able to yield very accurate results. In general, the success of applying the Monte Carlo method to solve a many-body problem depends on whether an efficient configuration generation algorithm is found. Usually an algorithm making use of local change may simplify the calculation of the hop rate R, but at the same time limits the speed of travelling between two vastly different configurations.

We have considered some possible modification and extension of the method including the N-particle jump procedure to go beyond the class with winding number zero and the approach based on the temperature independent degerenacy factors. The inclusion of the winding-number-1 class according to our algorithm does not lead to drastic changes in the results. The method via the degerenacy factor

proceeds along a different route and further investigations are required to ascertain its applicability and accuracy.

## Reference


Arfken, G. Mathematical methods for physicists (Academic Press)

Binder, K. 1979, in: Monte Carlo Methods in Statistical Physics, ed. K. Binder (Springer, Berlin)

Bonner, J.C., M.E. Fisher Phys Rev (1964) A135, 640

Bonner, J.C., H.W. Blöte, H. Beck and G. Müller 1981, in Physics in One Dimension (Springer, Berlin)

Data Processing Techniques, I.B.M., GC20-80110-0, 1980

De RAEDT, H. and Ad LAGENDIJK Phys. Rep. 127, No. 4 (1985), 233

Feynman, R.P. and A.R. Hibbs 1965, Quantum Mechanics and Path Integrals (McGraw-Hill, N.Y.)

Hirsch, J.E. and R.L. Sugra Physical Review B, V. 26 No. 9 1982

Hirsch, J.E. and R.L. Sugra Physical Review B, V. 26 No. 9 1982

Jordan, P. and E. Wingner Z. Phys. 47, 631, 1982

Metropolis, N., A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller and E. Teller 1953 J. CHem. Phys. 21, 1087

Suzuki, M. 1976a, Prog. Theor. Phys. 56, 1454

Suzuki, M. 1976b, Commun. Math. Phys. 51, 1833

Suzuki, M. 1976b, Commun. Math. Phys. 57, 193

Trotter, H.F. 1959, Proc. Am. Math. Soc. 10, 545